TEK SERVICE MANUAL
VOLUME 1

Part No.
070-4169-00

4027A
COLOR GRAPHICS TERMINAL

Please Check for
CHANGE INFORMATION
at the Rear of this Manual

First Printing OCT 1981

Tektronix®
COMMITTED TO EXCELLENCE

WARNING

THE FOLLOWING SERVICING INSTRUCTIONS ARE FOR USE BY QUALIFIED PERSONNEL ONLY. TO AVOID PERSONAL INJURY, DO NOT PERFORM ANY SERVICING OTHER THAN THAT CONTAINED IN OPERATING INSTRUCTIONS UNLESS YOU ARE QUALIFIED TO DO SO.

MANUAL REVISION STATUS

PRODUCT: 4027A Color Graphics Terminal

This manual supports the following versions of this product: B031651 and up

REV.	DATE	DESCRIPTION
@	OCT 1981	Original Issue

CONTENTS

Section 3 **CIRCUIT OVERVIEW**

Section 4 **DETAILED CIRCUIT DESCRIPTIONS**

ILLUSTRATIONS

TABLES

4173-100

Figure 1-1. 4027A Color Graphics Terminal.

Section 1

INTRODUCTION

The 4027A Color Graphics Terminal Service Manual is divided into two volumes. Volume 1 contains introductory information and theory of operation. Volume 2 primarily contains service information such as calibration procedures, parts lists and schematics. The purpose of Volume 1 is to provide the technical information necessary to understand how the terminal works. In addition, it can be used as an aid to isolating circuit malfunctions to a particular functional block. Detailed information on the use of the 4027A can be found in the following documents.

4027A Color Graphics Terminal Operator's Manual

4027A Color Graphics Terminal Programmer's Reference Manual

4027A Color Graphics Terminal Programmer's Reference Guide

GENERAL DESCRIPTION

The 4027A is a microprocessor-based raster scan computer terminal which has extensive color graphic capabilities. It can display alphanumeric and graphic data in up to 64 colors (eight at one time). Graphic commands display line segments and circles, as well as fill in areas with solid colors or user-defined patterns. Both alphanumeric and graphic data can be scrolled, simultaneously. The display can be divided into two separate areas, a "workspace" and a "monitor", each of which can be scrolled separately. A graphic input capability is provided, so that graphic data can be input directly to the display (and transmitted to an external device) by means of a graphic cursor. The terminal consists of a display unit and a detached keyboard and is designed for desktop operation.

SUMMARY OF CHARACTERISTICS

CRT: 13″ diagonal, high resolution, three color, delta gun with a 7.5″ (19 cm) by 10″ (25.4 cm) usable display area.

VIDEO BANDWIDTH: 15 MHz.

LINES/RASTER: 476 displayed, 525 scanned.

SCAN RATE: 30 Hz, interlaced.

REFRESH RATE: 30 times/second/dot, 60 times/sec/field.

CONVERGENCE: Independent adjustments in nine sectors, using an internally generated pattern.

FIRMWARE: System firmware enables the 4027A to respond to several dozen commands. These commands may be entered through the keyboard or may come from an external source such as a host computer. The commands instruct the terminal to perform such functions as creating graphics, defining colors, setting baud rates and so on.

COLORS: Sixty-four colors are possible, with eight colors displayable at one time. The eight displayable colors are designated C0 through C7. These colors, C0 through C7, can be redefined by command to any of the 64 possible colors.

CHARACTER GENERATION: 7 by 9 dot matrix displayed within an 8 by 14 dot cell.

CHARACTER SETS: The full 128 character ASCII set. Optionally, up to eight sets of 128 characters each (or 64 characters repeated twice). Math and ruling character sets are available in ROMs.

CHARACTER ATTRIBUTES: Within the workspace, characters may be displayed in any of the eight currently defined colors on the background color or inverted (background on foreground). Additionally, they can be made to blink from one color to another. Fields of characters can be protected so that they cannot be written over.

ALPHANUMERIC CURSOR: Wide Underscore.

DISPLAY SIZE: Thirty-four lines of 80 characters displayable. Can be scrolled through a buffer (Display Memory) of up to 32K bytes.

DISPLAY MEMORY: Contains a "display list" which specifies the content and structure of the display. 16K bytes is standard; 32K is optional. Assuming 40 characters per line of alphanumeric characters (no graphics) and 16K of display memory, the buffer will hold 10,240 characters (256 lines, or 7.6 pages). With 80 characters per line, under the same conditions, 11,680 characters (146 lines, or 4.2 pages) may be contained.

GRAPHICS: A graphics area can be defined in the workspace. Color graphics are drawn on a dot field (640 by 462 viewable) and are generated by a plotting algorithm which turns on on appropriate dots within an 8 by 14 dot matrix or graphic "cell". These cells are stored as "characters" in RAM and and are retrieved, as necessary to create the display. Vectors may be drawn and polygons may be filled and/or outlined with colors or user-defined color patterns.

GRAPHIC CURSOR: Full screen crosshairs controlled from the keyboard.

GRAPHIC INPUT MODE: Allows graphics to be drawn on the display using the graphic cursor. This input data can be simultaneously transmitted to an external device such as a host computer.

GRAPHICS MEMORY: Graphics cells are stored as characters in graphics RAM. One character requires sixteen 24-bit words. Standard complement is 16K words. Options are 32K, 48K, and 64K words.

SCROLLING: Alphanumerics and graphics can be scrolled up or down together.

SPLIT SCREEN: The screen can be divided vertically into two areas (workspace and monitor) which can be used for different purposes. For example, the workspace can be used for text editing while communication with the host computer takes place in the monitor. The two areas can be scrolled independently.

EDITING: Several special purpose keys on the keyboard allow local text editing.

KEYBOARD: Provides ASCII characters in standard typewriter configuration with a separate numeric pad and 16 edit and function keys. Key caps are replaceable for use with alternate character sets, and an overlay is available for relabeling the function keys. Firmware-controlled auto repeat and n-key rollover protection are provided.

PROGRAMMABLE KEYS: Most keys can be programmed to generate alternate characters or character strings (including commands).

STANDARD INTERFACE: RS-232 full duplex, 75 to 9600 Baud.

OPTIONAL INTERFACES: Half Duplex, Current Loop, GPIB.

VIDEO OUTPUT: Red, green, blue and monochrome video conforming to standard RS-330 is provided at BNC connectors on the rear panel. A separate multi-pin connector provides video and control signals for a 4612 or 4632 Video Hard Copy Unit.

COMPATIBLE PERIPHERALS: 4642 Printer (RS-232 port), 4632 Video Hard Copy Unit with Option 6 (Enhanced Gray Scale), 4923 Digital Cartridge Tape Recorder (RS-232 port), 4924 Digital Cartridge Tape Drive (GPIB interface), 4662 Interactive Digital Plotter (GPIB interface), remote video monitors.

Specifications

Environmental Specifications

The 4027A is intended to be operated in a stationry, indoor environment.

Temperature

Storage: —55 to +75 degrees C.
Operating: +10 to +40 degrees C.

Altitude

Storage: to 50,000 ft (16,000 m).
Operating: to 15,000 ft (4800 m).

Humidity

Storage: to 95%.
Operating: to 75%.

Physical Specifications

Weight

Keyboard: Approximately 5 lbs (2.3 Kg).
Display Unit: Approximately 95 lbs (43.2 Kg).

Dimensions: See Figure 1-2.

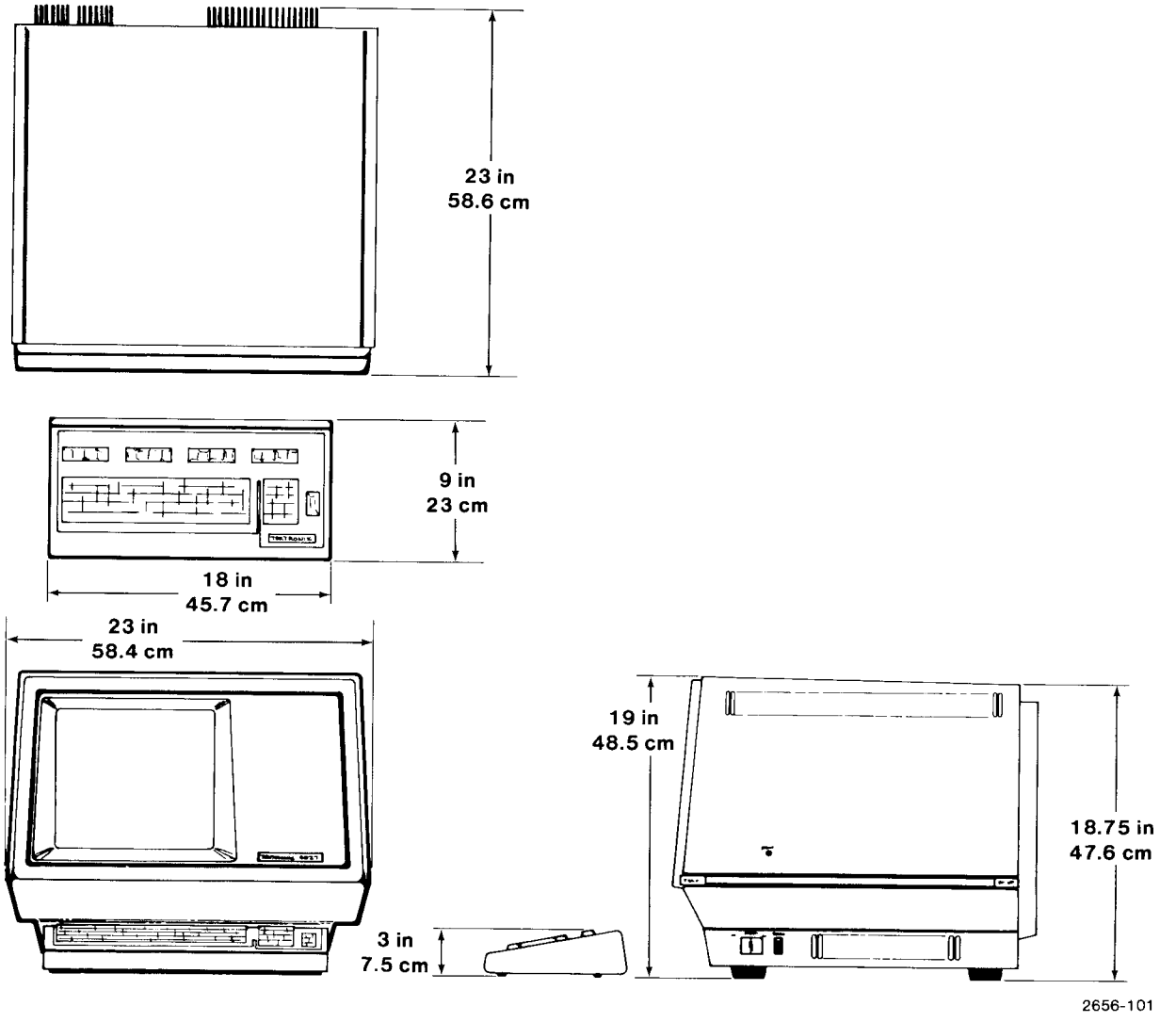


Figure 1-2. 4027A Dimensions.

ACCESSORIES

Standard Accessories

4027A Color Graphics Terminal Operator's Manual

4027A Color Graphics Terminal Programmer's Reference Guide

Blank keyboard overlay

Blank key cap (1x1)

Key cap cover (1x1)

Blank key cap (1x2)

Key cap cover (1x2)

Power cord

Optional Accessories

4027A Color Graphics Terminal Service Manual, Volumes 1 & 2

4027A Color Graphics Terminal Programmer's Referenc Manual

Alignment graticule

Extender board (for Convergence and Deflection circuit boards)

Extender board (for logic circuit boards)

SUMMARY OF OPTIONS

OPTION 1, HALF DUPLEX: Permits Half Duplex Normal and Half Duplex with Supervisor data communications. Consists one ROM chip and a Half Duplex cable.

OPTION 2, CURRENT LOOP: Provides 20 mA current loop interface. Consists of one circuit board, an "L" bracket, and current host cables.

OPTION 3, RS-232 PERIPHERAL INTERFACE: Permits communication with RS-232 peripherals. Option 36 must be installed. Consists of one circuit board and a cable.

OPTION 4, GPIB PERIPHERAL INTERFACE: Permits communication with the 4662/3 and 4924 peripherals. Option 36 must be installed.

OPTION 22, ADDED DISPLAY MEMORY: Increases the display memory size to 32K bytes.

OPTIONS 27, 28 and 29. ADDED GRAPHICS MEMORY: Increases the graphics memory to 32K, 48K and 64K words, respectively. (The word length is 24 bits.)

OPTION 31, CHARACTER SET EXPANSION: Permits up to seven alternate sets of 128 characters. Consists of one circuit board.

OPTION 32, 4027A RULINGS CHARACTERS: A 64-character line drawing set. Option 31 must be installed. Consists of one ROM chip.

OPTION 34, MATH CHARACTERS: A 64-character math set. Option 31 must be installed. Consists of one ROM chip.

OPTION 36, PERIPHERAL ROM: Provides the firmware to drive RS-232 compatible printers (with Option 3) and the 4924 Digital Cartridge Tape Drive or 4662 Interactive Digital Plotter through the GPIB (with Option 4). Consists of two ROM chips.

HARD COPY AND VIDEO OUTPUT: A 15-pin connector provides output to the 4632 Video Hard Copy Unit. Six BNC connectors provide the following signals: monochrome composite video, red, blue and green composite video, composite sync and blanking.

OPTION 4A, UNITED KINGDOM CHARACTER SET: Permits terminal to display the pound sign used in the United Kingdom ( ) instead of the pound sign used in the U.S. (#). Consists of one ROM and one keycap.

OPTION 4B, FRENCH CHARACTER SET: Changes terminal's keyboard to the French "AZERTY" configuration by changing four keys. Consists of one ROM and four keycaps.

OPTION 4C, SWEDISH CHARACTER SET: Permits terminal to display Swedish characters. Consists of two ROMs and seventeen keycaps.

OPTION A1, 220V/16A 50Hz Operation Universal European plug.

OPTION A2, 240V/13A 50Hz Operation United Kingdom plug.

OPTION A3, 240V/10A 50Hz Operation Australian plug.

OPTION A4, 240V/15A 60Hz Operation North American plug.

Section 2

GENERAL DISPLAY CONCEPTS

4027A RASTER SCAN DISPLAY

The 4027A uses a 525 line, 60 Hz interlaced raster scan display. In this system, the electron beam scans the crt face at 15750 Hz in the horizontal direction and at 60 Hz in the vertical direction. There are exactly 262.5 horizontal scans for each vertical scan. This results in an interlaced scanning pattern or "raster" (see Figure 2-1). The video information to be displayed is presented in two sequential "fields." One field consists of the information for all the even raster lines. The next field consists of the information for all the odd raster lines. The two fields taken together constitute a "frame". The frame consists of two complete vertical scans and 525 horizontal scans; it occurs 30 times a second.

As shown in Figure 2-1, the field beginning at point A contains all the odd lines down to point B at the middle of line 483 (a total of 241.5 scan lines). The beam travels from point A to the right side of the screen in about 54.5 μs. It then flies rapidly back to the left side of the screen. This horizontal retrace takes about 8 μs during which the electron beam is turned off. The electron beam travels down the face of the screen at the vertical scanning rate. Upon reaching point B, the electron beam is turned off and a vertical retrace begins.

The vertical retrace takes exactly 21 horizontal scan periods. Horizontal scanning still goes on, but since the electron beam is turned off, it is not visible. At the end of this period, the electron beam arrives at point C where a new vertical scan begins. The electron beam covers the field from point A to point B to point C in 1/60th of a second. The second field is scanned in a similar fashion except that it begins at point C, goes to point D, then back to A. Like the first field, it takes 1/60th of a second and consists of 262.5 total horizontal scans.

Raster lines 1 through 476 are used for displaying graphics and text. Lines 477 through 483 are not used. The half line beginning at pont C is also unused and is blanked off.

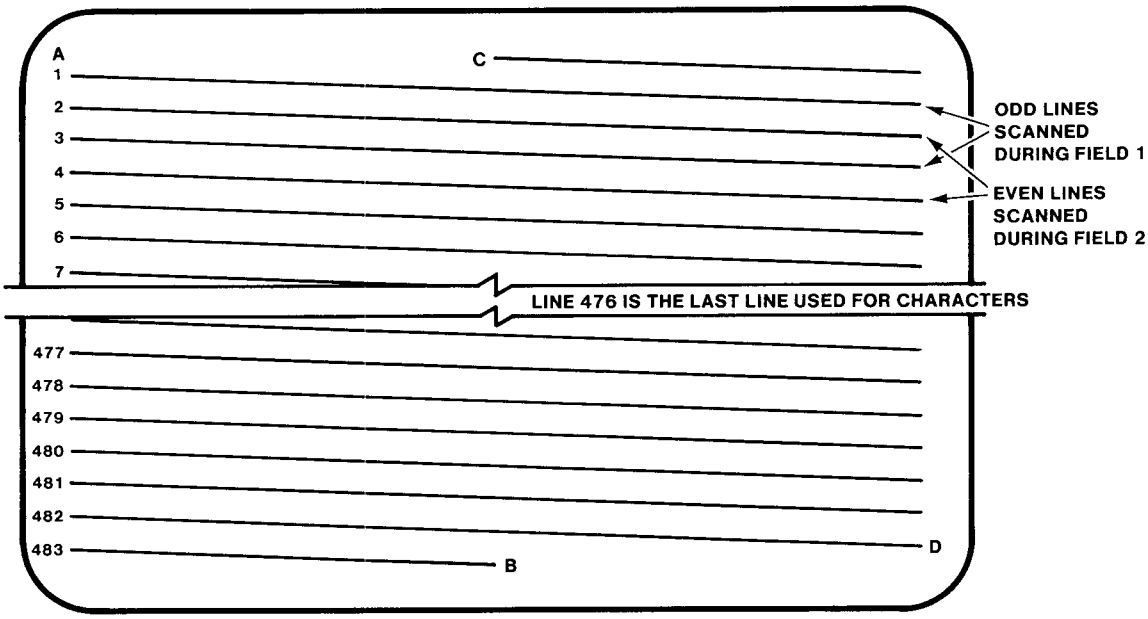


Figure 2-1. 4027A Raster Scan Display.

THE PIXEL

All display information is conveyed by turning the electron beam on and off as it scans the screen. The smallest unit of information in the display is the "pixel" (short for picture element). A pixel is produced by turning the electron beam on or off for 81.4 nanoseconds during a raster line. This produces a dot on the face of the crt. In the 4027, pixels can be one of 64 different colors, including black. The mechanism for producing these colors is discussed in a later section.

Figure 2-2 depicts how a display can be made up from pixels. Notice that in order to produce three adjacent pixels, the electronic beam is turned on for 244.2 ns.

Information about which pixels in the display are "on" and "off" is stored as binary data within a memory. This data is recalled from the memory and used to turn the electron beam on and off as it scans the screen.

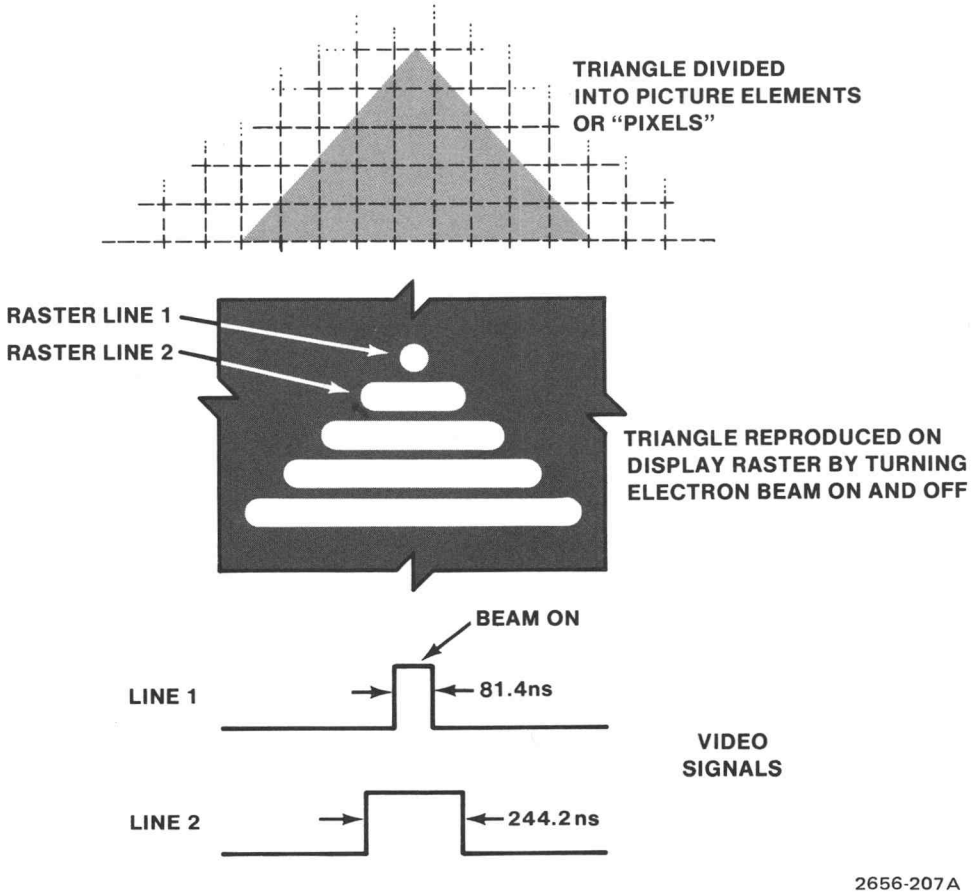


Figure 2-2. Constructing a Display From Pixels.

CHARACTERS

For purposes of memory storage and processing, pixel data is organized into character cells. The character cell is an 8 wide by 14 high block of pixels (a dot matrix) used for making characters. Characters, in turn, are used to make up the display.

As shown in Figure 2-3, a character may be a single symbol, such as a letter, or it may be part of a larger "graphics" figure such as a triangle or circle.

The alphanumeric characters (and certain special symbols), which are used to make up text, are stored as bit patterns in Read Only Memory (ROM). A single ROM may contain the bit patterns for a "character set". For example, the alphanumeric characters used in the 4027A are referred to as the "ASCII character set" and are stored in the ASCII ROM. Pixel data for a ROM-contained character occupies 16 eight-bit bytes of character memory (although all 16 bytes aren't used).

Graphics characters are made up by the terminal as they are needed to produce graphics figures. For this reason, they are stored in Random Access Memory (RAM). A graphics character cell occupies *three* 16-byte blocks of character memory space; three bits of data are stored for each pixel. The three bits are needed to select a color for the pixel.

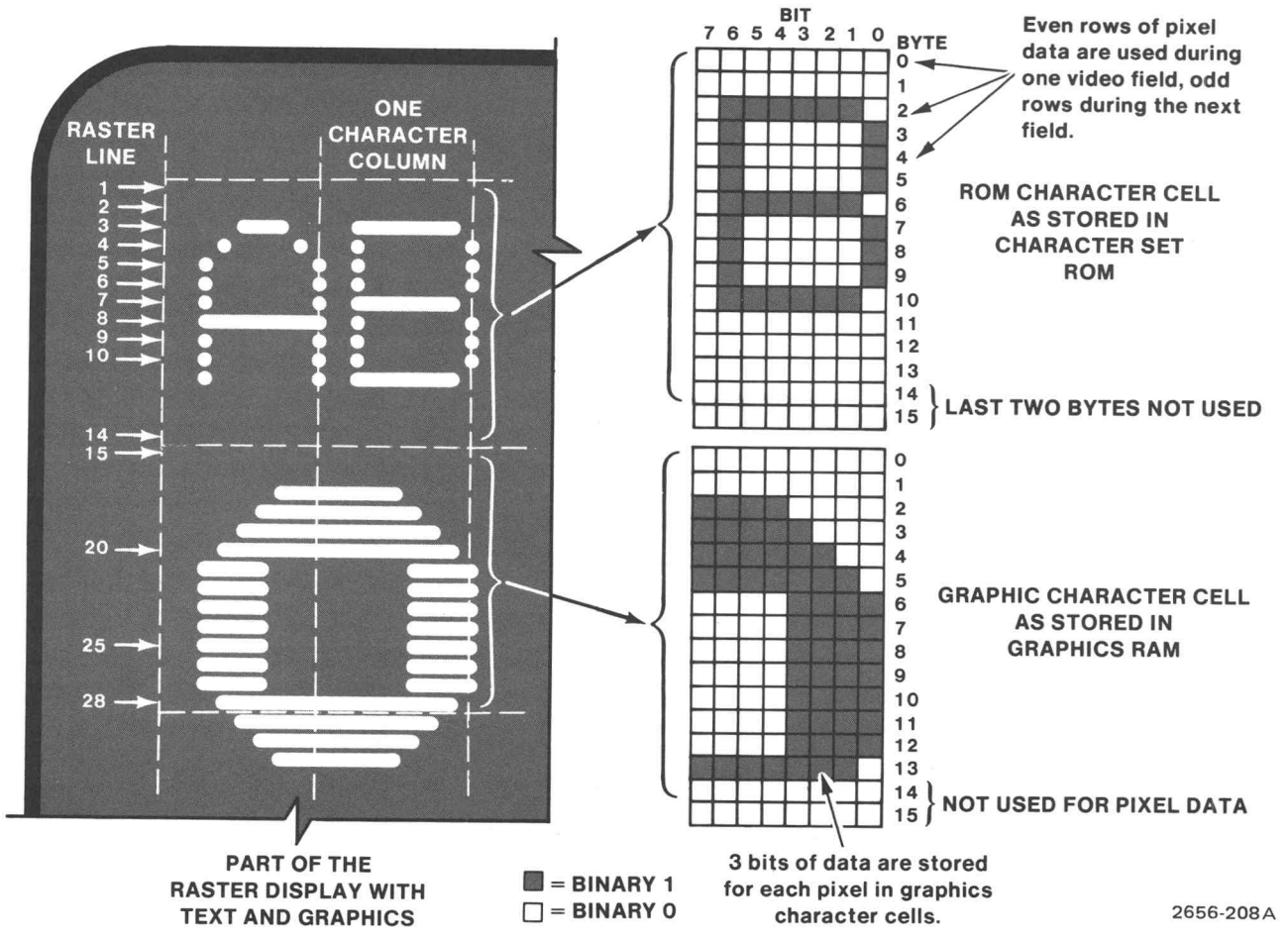


Figure 2-3. 4027A Character Cell.

THE DISPLAY FORMAT

The display consists of 34 rows of 80 characters. Each row of characters takes 14 raster lines, 7 in the even video field and seven in the odd field. Raster line 476 forms the bottom of character row 34. Alternatively, the display can be thought of as a matrix of pixels 640 wide by 476 high on which graphics can be drawn. See Figure 2-4.

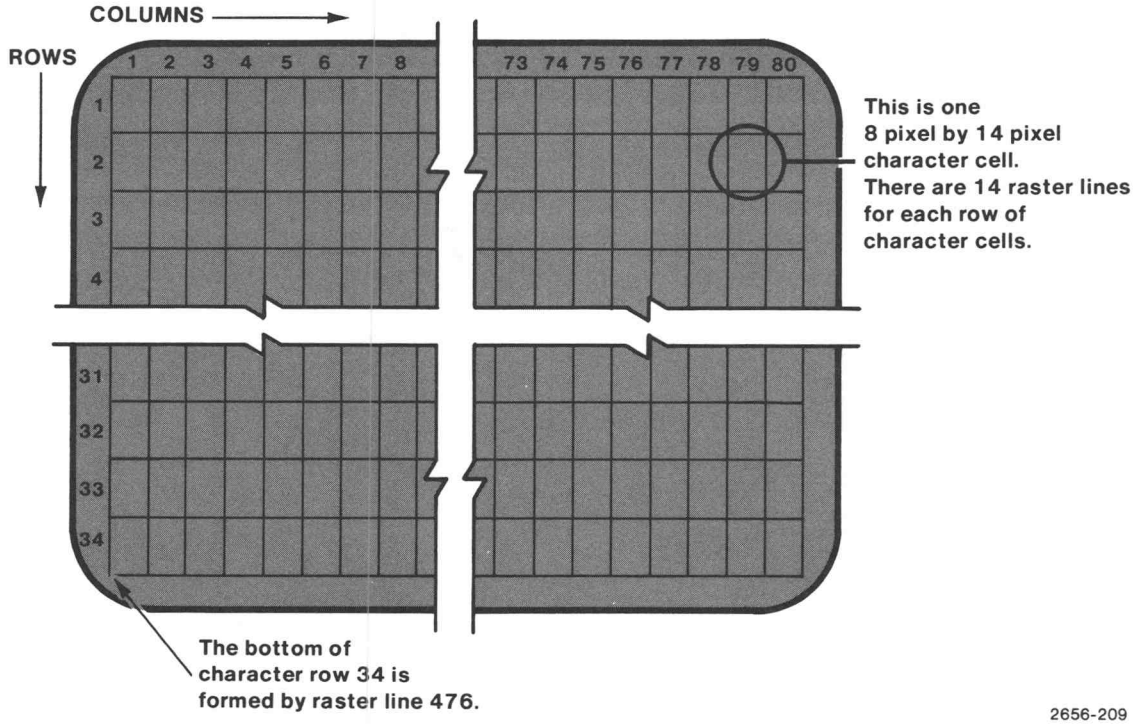


Figure 2-4. Display Format.

THE COLOR SYSTEM

Figure 2-5 depicts the system which is used in the 4027A for producing color displays. The color crt contains three electron guns. There are also three phosphors on the crt faceplate; one produces red light, one green light, and one blue light. Each electron gun excites only one of the phosphors. The beam current in each electron gun (and therefore, the intensity produced in the phosphor it excites) is controlled by a Digital to Analog Converter (DAC). A 2-bit binary input to each DAC selects one of four discrete beam current levels for each electron gun, including off. All display colors are combinations of the resulting four levels of intensity of red, green, and blue. Since each color has four different brightness levels, there is a total of 64 possible colors.

Out of these 64 possible colors, however, the terminal can select only eight to be displayed at one time. The eight selected colors are assigned to "color numbers" C0 through C7, and all color graphics are specified in terms of these numbers. The color numbers are, in fact, locations within a small RAM called the Color Map. The assignment of colors to the eight color numbers is determined by the terminal's microprocessor (acting on certain commands from the terminal's command set). At each Color Map address, the microprocessor loads six bits of data: two bits each for red, green, and blue. This data drives the DACs which control the crt's electron guns.

In the case of graphics characters, a particular color map address is selected by the pixel data that is stored in graphics RAM. There are three bits of data stored for each pixel. These three bits compose an address to the Color Map which selects a color for the pixel.

In the case of ROM-contained characters, only one bit of data is returned from character memory for each pixel. This determines only whether the pixel is off (background) or on (foreground). The color map addresses for the ROM characters are selected by visual attribute codes. These codes come from a different source than the pixel data; they determine the color of the entire character, rather than individual pixels within the character.

The single graphics character depicted in Figure 2-5 has three small polygons drawn in it. Assume that the middle polygon is yellow. The color data which produces yellow on the display screen has been loaded into location 4 in the Color Map. This means that the pixel data for the yellow polygon must select location 4. Taking plane A as the most significant bit, and plane C as the least significant bit, the data returned from character memory for that pixel must be 1, 0, and 0.

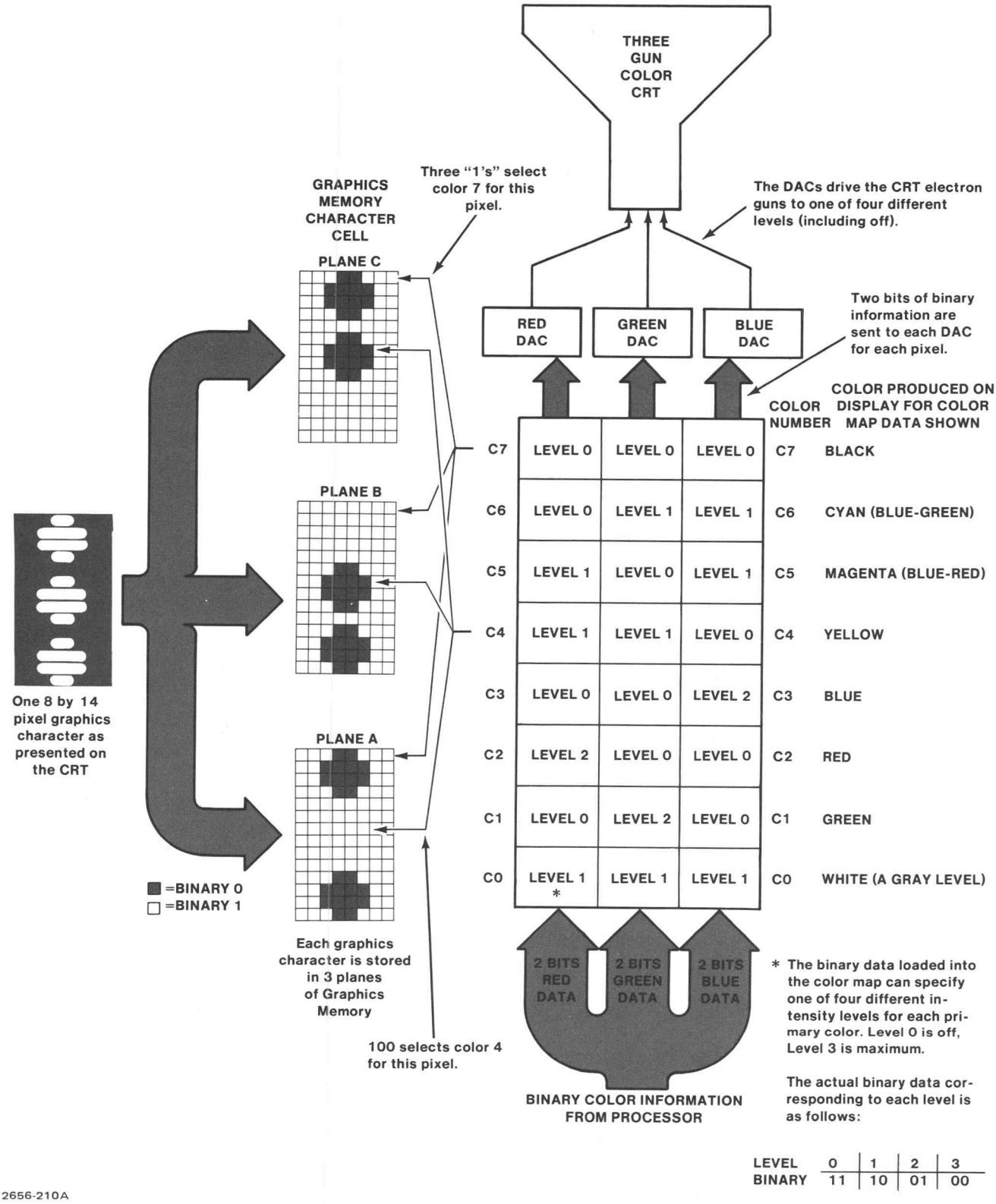


Figure 2-5. 4027A Color System.

VIRTUAL BIT MAP AND DISPLAY LIST CONCEPTS

A bit map is picture information stored as a collection of binary digits in a memory device. For a simple black and white display there might be one memory bit for each pixel on the display screen. If the bit contained a 1, the pixel would be on; if a 0, the pixel would be off. However, this requires a lot of memory, even for simple displays, since data must be stored for every pixel, whether it is off or on.

The 4027A uses virtual bit mapping. The term "virtual" implies that this kind of bit map is an "apparent" bit map. A virtual bit map display appears to be bit mapped, but memory is not required for most of the blank spaces which appear in a typical display. In a sense, virtual bit mapping allows the display to be "compacted" before it is stored in memory.

As shown in Figure 2-6, the 4027A's display is built up character row by character row from a display list. The display list specifies which characters are in each character row. It contains the character memory addresses of the pixel data for these characters. If there are only three characters in a particular row, the display list need only contain three character addresses and an end-of-line marker for that row.

Using a display list saves memory space in two ways. First, it isn't necessary to store data for trailing blank spaces in unfilled character rows. Secondly, pixel data for characters that are often used is only stored once in character memory. For example, the pixel data for the interior of a solid-colored graphic figure, like a polygon, may be stored as one character cell in graphics RAM. Likewise, the text character "A" is stored in Character Set ROM and need only be referenced by address in the display list.

The following points summarize the method used in the 4027A for generating a virtual bit map display (refer to Figure 2-6):

- A display list is stored in display memory. The display list contains character addresses. These addresses are the locations in character memory where pixel data is stored for all the text and graphics in the display. The display list also contains "attribute" codes and the alphanumeric cursor's location. The attribute codes control the manner in which non-graphics characters are displayed (color, inverted/normal video, etc.).

- The character addresses for one row of text are extracted from the display list and stored in a row buffer. The row buffer is a temporary memory which has a location for each of the 80 character positions in the row. If a row has less than 80 characters in it, the remainder of the buffer is filled out with ASCII spaces.

- There are two row buffers. As one row buffer is being filled with addresses from the display list, the other supplies addresses to the character memories. Each row buffer is read seven times for a character row. This causes the character memories to return seven sequential line of pixel data for each character row.

- The pixel data is a series of data bytes one character cell (eight bits) wide. In the case of graphics characters, three parallel bytes of data are needed for each row of pixels within the character cell. For ROM characters, only one byte is needed for a row of pixels.

- Each byte of pixel data is loaded into a shift register. It is then sent bit-serially to the video circuits where it modulates the electron beam in step with the display raster.

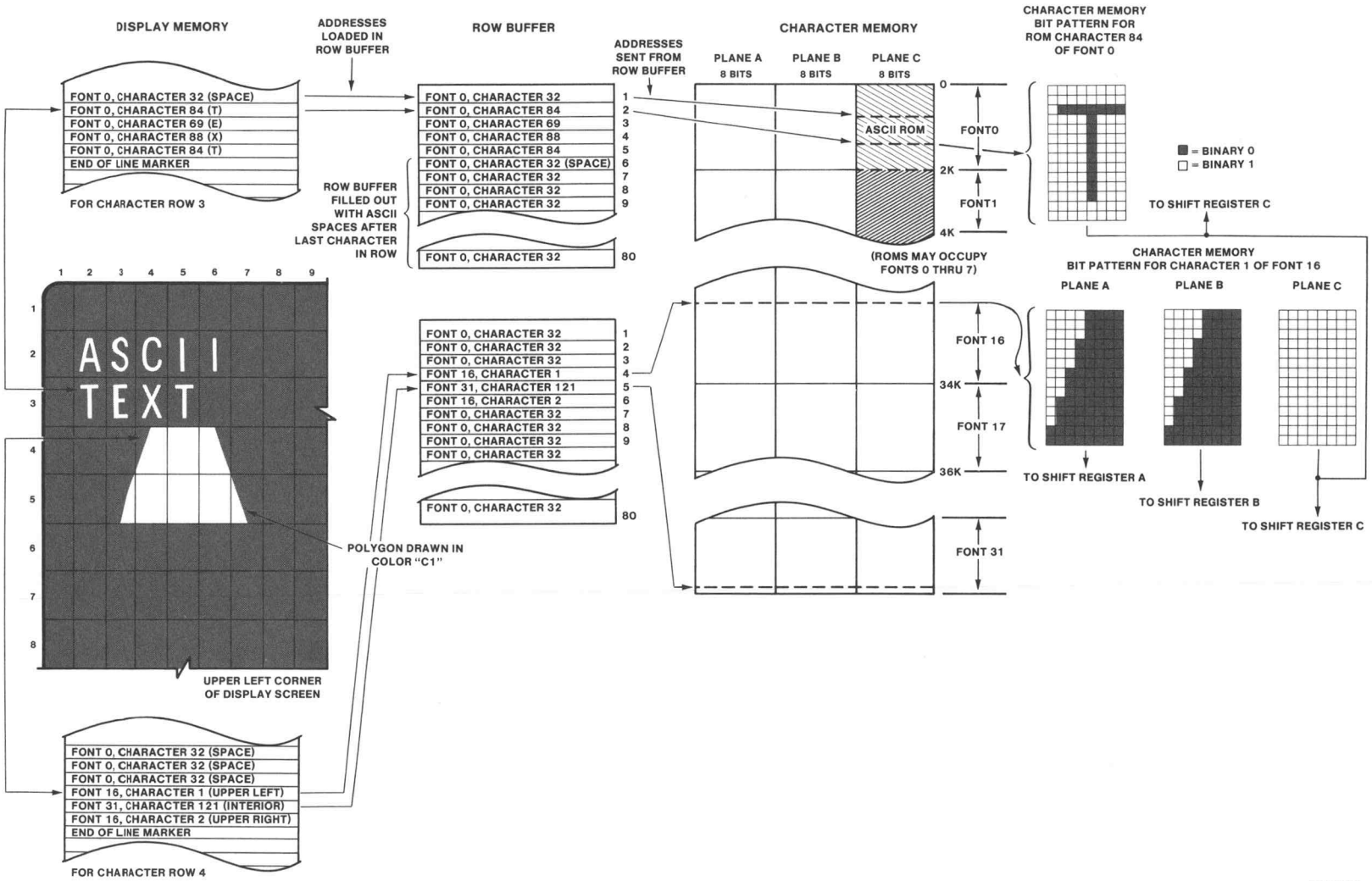


Figure 2-6. 4027A Virtual Bit Map and Display List Concepts.

Section 3

CIRCUIT OVERVIEW

The purpose of this section is to provide an introduction to the terminal's circuitry. Figure 5-1, the 4027A System Block Diagram depicts the functional interconnection between all the standard circuit boards that make up the terminal. It also shows the approximate physical locations of the circuit boards with respect to one another. This should be used along with Figure 5-2, the Display Processor Block Diagram, and Figure 5-11, the Color Video Display Block Diagram, during the following discussions.

THE TERMINAL'S FUNCTIONAL ORGANIZATION

In terms of function, the terminal may be divided into three broad areas: Display Processor, Color Video Display, and Communications. Figure 3-1 shows these major functional blocks and the flow of information between them.

Display Processor

The Display Processor coordinates the terminal's functions and assembles, stores, and modifies all the information to be presented on the display screen. Its major subunits are the Microprocessor, the System Memory, the Display Memory, the Display Controller, and the Character Memory (Graphics RAM, Character Set ROM).

The Microprocessor, controlled by firmware contained in the System Memory, transfers data between the communications interfaces and the transmit and receive buffers in display memory. It sets up and modifies the display list in Display Memory, and makes up and stores graphics character cells in the Graphics Memory portion of the character memories. It also controls the display colors and the position of the graphics cursor through the Display Controller.

The Display Controller receives character addresses and visual attribute codes from the Display Memory, uses these to retrieve the appropriate pixel data from the character memories, and converts this data to a serial video data stream. The video data is sent, along with horizontal and vertical sync signals and a cursor signal, to the Color Video Display.

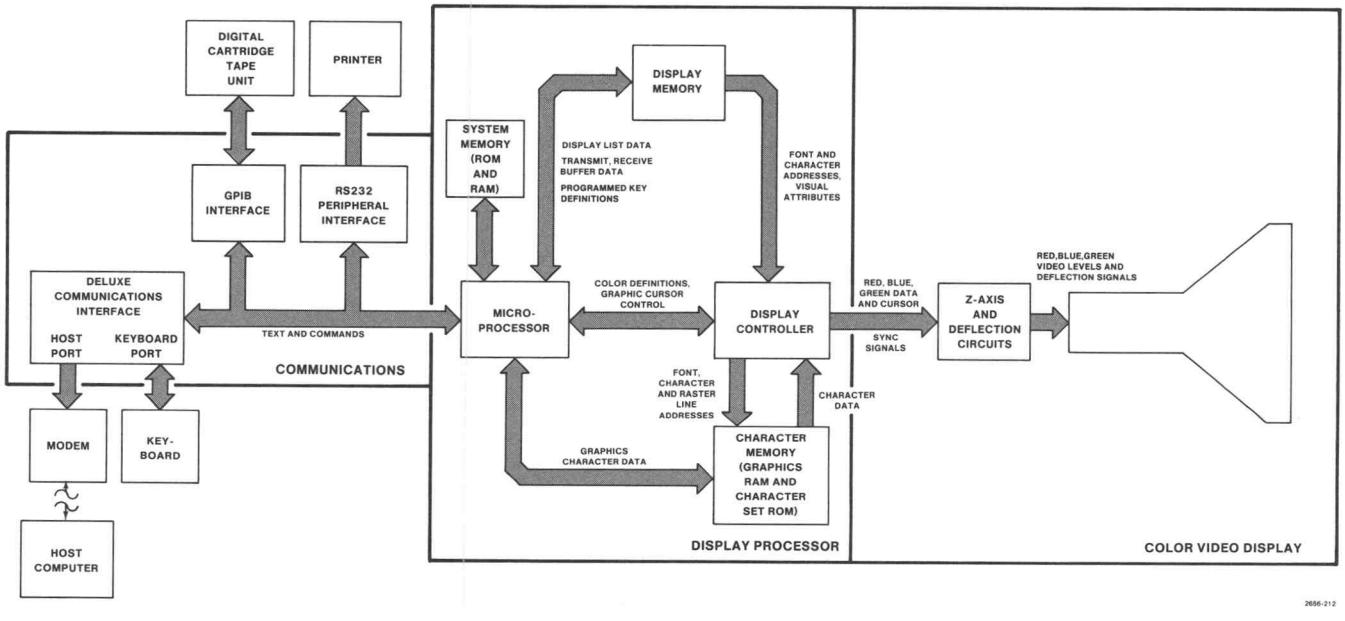


Figure 3-1. 4027A Functional Block Diagram.

Color Video Display

The Color Video Display converts red, green, and blue video data coming from the Display Controller into the appropriate drive levels for the electron guns in the crt. It uses the horizontal and vertical sync signals to trigger synchronous deflection signals for the display raster (see Table 3-1).

Table 3-1

HARD COPY AND VIDEO OUTPUTS

Labeled	Jack No.	Cable Color	Description
MONOCHROME	J5300	white	monochrone video
RED	J5600	red/white	red composite video
GREEN	J5700	green/white	green composite video
BLUE	J5800	blue/white	blue composite video
SYNC	J5500	orange/white	composite sync
BLANK	J5400	brown/white	composite blanking

Communications

The standard communications interface for the terminal is the Deluxe Communications Board. This unit offers two ports: one for the keyboard and one for host communications. The host port transmits and receives RS-232 asynchronous serial data and may operate in either full duplex or half duplex (optional) modes. An optional GPIB interface allows the terminal to communicate with the TEKTRONIX 4924 Digital Cartridge Tape Unit and 4662 Interactive Digital Plotter. An optional RS-232 interface allows the terminal to send data to the TEKTRONIX 4641 and 4642 Printers.

DISPLAY PROCESSOR

See Figures 5-1 and 5-2.

Display Memory Board

Display RAM

The Display RAM is a multipurpose block of memory occpying the upper 32K of the Microprocessor's address space. It contains the display list, the transmit and receive buffer, and programmed key definitions.

Tracker

The Tracker circuit scans the display list and passes on font addresses, character addresses, visual attributes, and cursor location to the Display Controller. The Tracker is controlled by tracker commands embedded within the display list. These commands tell it where to begin scanning the display list, what kind of information it is reading (font, character, etc.), and how to link all of the information together. The Tracker sends the Tracker Clock to the Display Controller. This is used to synchronize loading of the display list data into the Row Buffers.

Display Controller Board

Tracker Data Decoder and Latches

As the 8-bit bytes of display list data come from the Display Memory, they are examined by a decoding circuit. If a character address is present, it is loaded directly into one of the Row Buffers. If a font address or visual attribute code is present, it is first stored in a latch. Then it is loaded into the Row Buffer in parallel with the character addresses to which it applies.

Row Buffers

There are two Row Buffers. Each Row Buffer is in effect a 16-bit by 81-word memory. While one Row Buffer is being loaded with display list data, the other Row Buffer supplies its addresses to the character memories. The Row Buffer which is supplying addresses to character memory is clocked by the Column Clock signal. Column Clock occurs in step with the display raster and causes one address to be sent at the proper time for each of the 80 character columns. The first byte in the Row Buffer is the alphanumeric cursor location. After the cursor, the next 80 addresses contain a 16 bit word specifying font address, character address, and visual attribute.

Raster Address Counter

Part of the address information used by the character memories is the raster line address. This specifies which line of pixel data within the character cell is needed. During each raster field, every other raster line is scanned. This means that during one field the seven even lines of pixels are scanned, and on the next field, the odd lines are scanned. The Raster Address Counter supplies odd addresses (1, 3, 5, etc.) during one field and even addresses during the next. The Row buffer is read seven times for each character row. It supplies the same characters, fonts, and attributes each of the seven times; only the raster address changes.

Shift Registers A, B, and C

The data for graphics character cells is returned to the Display Controller as three parallel 8 bit bytes; one each from Planes A, B, and C of Graphics Memory. The eight bit bytes are loaded, in parallel, into three shift registers. Once loaded, they are shifted out serially by a 12 MHz clock signal generated by the Timing and Control Logic. The three serial outputs of the shift registers combine to form a three-bit address sent to the Color Map Address Multiplexer. This three bit address selects a color from the Color Map for each pixel in the graphics character cell.

For ROM characters, which return only one byte of pixel data at a time, only Shift Register C contains meaningful information. Its output serves to switch the Color Map between background and foregound color. The choice of background and foreground color is determined by visual attribute codes from the display list.

Color Map Address Multiplexer

The Color Map Address Multiplexer steers a three-bit address from one of the following four sources to the Color Map:

- Source 0. This source is selected when displaying color graphics. The address consists of the three serial outputs of Shift Registers A, B, and C. Source 0 is always selected whenever RAM data is present on the data lines from Graphics RAM.

- Source 1. This source is selected during retrace periods. It consists of the outputs of the Color Map Address Latch. During vertical retrace, the Processor can address the Color Map and change the color definitions it contains.

- Source 2. This source is always binary 111; it selects color 7 in the Color Map. Color 7 is the standard background color.

- Source 3. This source consists of the low order bits of the visual attribute codes coming from the Row Buffers. These three bits determine which of the eight colors in the Color Map will be used to display the standard ROM characters. Whenever ROM data is returned from Character Set ROM, the serial output of Shift Register C causes the Color Map Address Multiplexer to switch between Source 2 and Source 3 (i.e., between the background and foreground). A fourth visual attribute bit, when true, causes the sequence of switching to be reversed. This displays the characters with inverted video (the characters are displayed in color 7 on a background determined by the visual attribute code).

Color Map

The Color map is a small RAM which is six bits wide by eight words long. The data which determines the actual color of each pixel is stored here. A three-bit address from the Color Map Address Multiplexer selects one of the eight words stored in the RAM. Of the six bits of data stored at each location, two bits control the intensity of the red electron gun, two bits control the green electron gun, and two bits control the blue electron gun. Sixty-four combinations of red, green, and blue are possible. However, since only eight locations can be selected in the Color Map, only eight colors are displayable at one time. The Processor must change the data in the Color Map when a different color is needed within the eight addressable categories (this can occur only during retrace periods).

Graphics Memory Board

The Graphics Memory Board holds the RAM where graphics character cells are stored. Graphics Memory is 24 bits wide. Each 24-bit word is divided into three sections (Planes A, B, and C) of eight bits apiece. The array may be as long as 64K words. It is divided into 32, 2K sections termed "fonts". Graphics RAM lies entirely on the Graphics memory circuit board. All of it addressing and control functions are performed on the Graphics memory Controller board.

Character Set ROM

Character sets such as the ASCII characters and the optional ruling characters are contained in ROM. The ASCII ROM lies on the Graphic Memory Controller board. All additional ROM lies on the Character Set Expansion board. The ROM array may be as large as 16K bytes. Like graphics memory, it is organized into 2K fonts. It occupies the lower eight fonts in character memory address space. ROM and RAM may overlap in address space. In this case, ROM takes precedence over RAM. All address decoding and control functions for the ROM are done by the Graphics Memory Controller Board.

Graphics Memory Controller Board

The Graphics memory Controller performs three major functions. First, it receives font, character, and raster addresses from the Display Controller and passes back the corresponding character cell data synchronously, as it is needed for the display. Second, it allows the processor to read the character memories and write character cell data into Graphics RAM as it creates graphics. Third, it refreshes the dynamic RAM devices periodically.

ROM data passes over the same data paths as the Plane C RAM data. Routing of the data is performed by the address decoding logic. The Processor accesses the character memories by means of several I/O ports in the Processor's address space.

Processor Board

The Microprocessor

The Microprocessor directs more of the terminal's operations. To summarize the most important ones: It transfers characters between the communications interfaces and the transmit and receive buffers in Display memory. It makes and stores graphics character cells in Graphics memory and reads the character memories to determine what is there. It reads and modifies the display list as text and graphics are added to the display. It determines which colors from the terminal's 64 possible colors are assigned to the eight color numbers by loading data into the Color Map. It controls the position of the alphanumeric and graphic cursors. It recognizes and acts on the terminal's command set.

The Microprocessor can directly address 65,384 memory locations. Its address space is divided into three major sections: System Memory, which contains programs that control the Microprocessor itself; Display memory, which is the display list buffer for the terminal; and the I/O ports, by which the Microprocessor communicates with many of the functional units of the terminal.

In addition to the directly addressable memory space, the microprocessor can indirectly address additional System Memory through a "paging" scheme. It can also access the entire character memory through an I/O port.

System Memory

The System Memory occupies the lower half of the terminal's address space. It consists mostly of ROMs containing the firmware which controls the Microprocessor. However, there is a small amount of System RAM, used for temporary storage, as well as a small battery powered RAM which is used to retain certain settings, such as baud rates, during power-off periods. The Processor board contains 32K bytes of the System ROM; additionally, there is 24K bytes of space for options.

COMMUNICATIONS

The communications interfaces perform all data transfers between the terminal and external devices. All communication is in the form of ASCII characters. The standard interface for the terminal is the Deluxe Communications board. This board offers two ports to external devices: the host port and the keyboard port. The host port transmits and receives RS-232 serial data in asynchronous, half duplex and full duplex formats. The keyboard port receives ASCII characters from the keyboard and transmits certain control and status signals to it.

Two optional interfaces permit communication with additional RS-232 and GPIB devices. These are the RS-232 Peripheral Interface board and the GPIB Interface board. They are used with the TEKTRONIX 4642 Printer and 4924 Digital Cartridge Tape Unit, respectively.

COLOR VIDEO DISPLAY CIRCUITS

See Figures 5-1 and 5-11.

Deflection Board

The primary purpose of the Deflection board is to generate the horizontal and vertical deflection currents which drive the deflection yoke. It also produces the DYNAMIC FOCUS signal, the H.V.INHIBIT (High Voltage Inhibit), and the two voltage ramp signals, HRAMP (Horizontal Ramp) and VRAMP (Vertical Ramp).

The horizontal and vertical circuits are synchronized to HSYNC-0 and VSYNC-0, respectively. These two signals are produced by the Display Controller board, synchronously with the character data it is sending to the Z-Axis circuits.

HRAMP and VRAMP are used in the deflection circuits for geometry correction (to make the raster appear rectangular on the crt screen) and on the Convergence and Z-Axis board to generate the convergence waveforms.

The DYNAMIC FOCUS signal is used in the Focus Supply to modulate the focus potential in step with the deflection signals. This keeps the electron beam in focus over the entire crt screen.

H.V.INHIBIT shuts down the high voltage circuits. In the event that horizontal or vertical deflection is lost, this prevents damage to the crt.

Convergence and Z-Axis Board

The convergence circuits drive the convergence yoke to produce small independent corrective deflections for each of the three electron beams. This is required so that the three beams converge on one spot as they are deflected across the face of the crt. HRAMP and VRAMP are transformed by the convergence waveform generator circuits into three corrective signals: HCONV (Horizontal Convergence), VCONV (Vertical Convergence), CCONV (Corner Convergence). These three signals are then divided into components which correspond to eight different areas on the crt screen. This allows independent adjustment of convergence in these eight different areas and in the center of the screen.

The Z-Axis circuit receives 6 bits or red, blue and green data for each pixel from the Display Controller and converts this data to beam current levels in the crt. There are two bits of data for each electron gun. These two bits select one of four levels of beam current, including off. The four levels of red, blue, or green intensity combine to produce the terminal's 64 colors.

High Voltage Supply

The High Voltage Supply consists of two circuit boards: the High Voltage Control board and the High Voltage board. The High Voltage Control board produces a 250 volt, 30 KHz sine wave voltage; this voltage is stepped up by a transformer and voltage multiplier on the High Voltage board to produce the 21 KV anode potential for the crt. A sample of the 21 KV anode potential is fed back to the High voltage Control board where it controls the amplitude of the 30 KHz sine wave, thus stabilizing the high voltage output.

The High Voltage board produces a 6 KVDC focus potential which has the DYNAMIC FOCUS signal superimposed on it. In addition, three adjustable screen grid voltages are produced for the three electron guns.

HOW THE 4027A CREATES A TYPICAL DISPLAY

The following discussion presents a comprehensive example which ties together many of the concepts presented earlier. In addition, it demonstrates the most important interactions between the terminal's firmware and its hardware.

Figure 3-2 shows a simplified 8-column by 4-row display. It also shows a display list for that display and the path of the Tracker through the display list. Initially, the Processor sets up one 16-byte block of display list for each character row in the display. At the beginning of the display list there is a pointer block containing Tracker jump instructions which tell it to jump to the beginning of the "workspace" and "monitor." These jump instructions in effect divide the display list into two sections which can be used independently. In Figure 3-2, no workspace has been defined yet, so the display list is all monitor. The pointer block also contains, near its end, a loop where the tracker waits during vertical retrace periods. At the end of vertical retrace, the Tracker is reset to the starting location of the display list (which is always X'F000').

The Tracker follows a rather devious path through the display list. Notice that all the 16-byte blocks are linked by jump instructions. Since there are no text or graphics on the display yet, the Tracker is passing only ASCII space characters to the Row Buffers.

The cursor is in the upper left corner of the display. Its location is specified by the Processor. To do this, the Processor loads a number into byte 4 of the beginning display list block for each character row. In Figure 3-2, byte 4 of block specifies that the cursor appears in column 1. For the other character rows, the specified location is column 9. Since there are only eight columns in the display, the cursor does not appear on these rows.

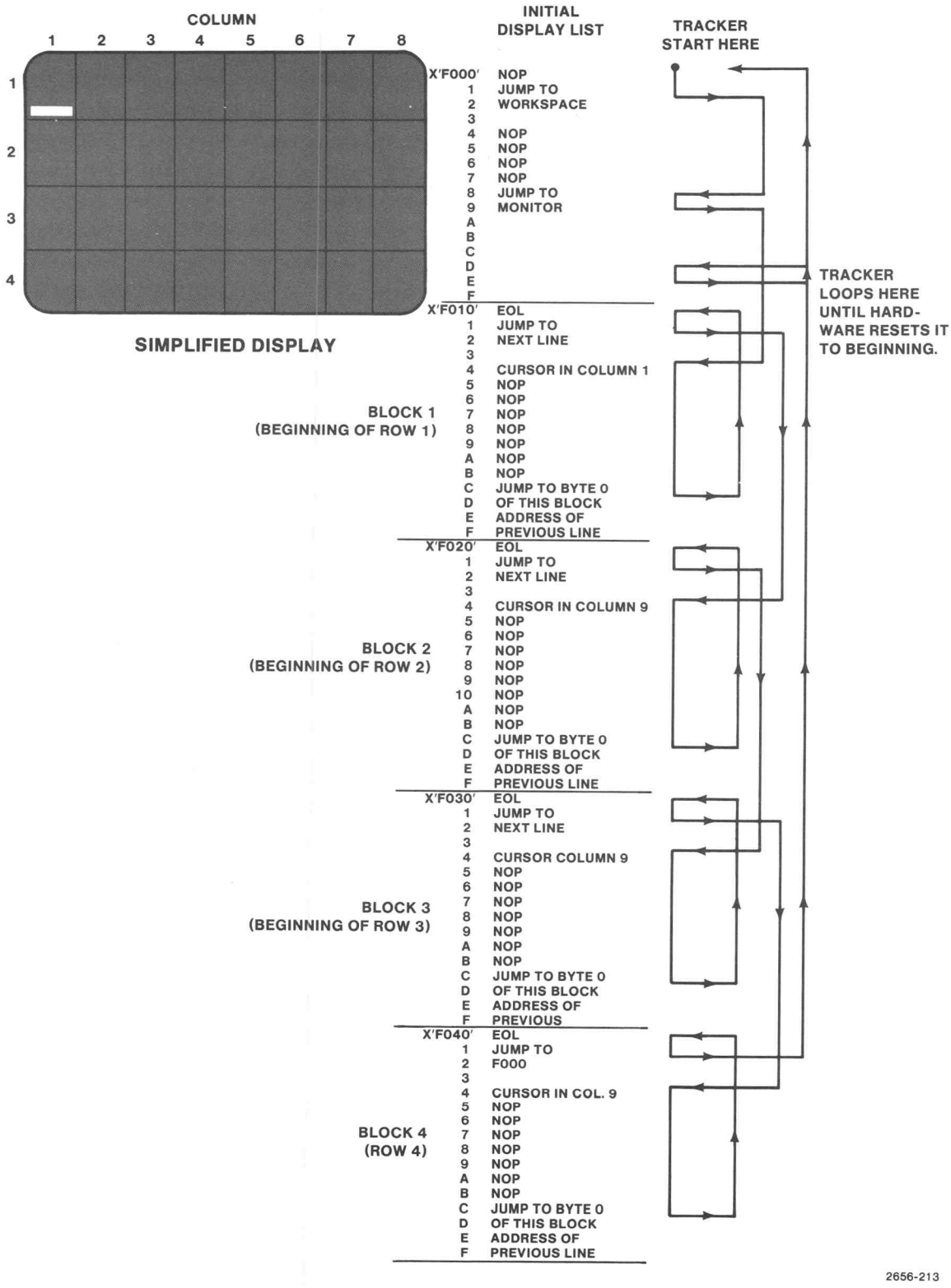


Figure 3-2. Simplified Display and Display List.

Figure 3-3 shows the same simplified display, but with some text and graphics added. The following points (labeled A, B, C, etc. on the figure) summarize the hardware-firmware interactions which take place in producing the display.

A. Text and graphics character addresses are added to the display list. There aren't enough bytes in block 1 to hold all of the character addresses for row 1, so the processor finds a vacant block (block 5) of display RAM and adds the "s" from the end of row 1 there. In the actual 80 column display, it could take many more blocks to hold all of the characters in one row. They could be spread all through the Display Memory, linked together only by the jump instructions.

B. The characters "TEXT" at the beginning of row 2 are displayed with the "inverted" visual attribute. To specify an inverted attribute, the processor inserts an appropriate attribute code (ATT I) in block 3, just before the characters to which it applies. The attribute code is loaded in parallel with the character and font addresses in the Row Buffers.

C. A graphics area, consisting of columns 5, 6, 7, and 8 in rows 2 and 3, has been defined. When the terminal receives a command to set up a graphics area, the Processor sets aside enough blocks in Display Memory to hold addresses for all the character cells in the graphics area. for the simplified display shown, only eight character cells are needed. Blocks 6 and 7 take care of these.

   To draw the solid triangle shown in the graphics area, first the color, then the three vertices are specified. The color is specified by one of the eight color numbers (C0 through C7). The vertices are specified in terms of X and Y coordinates which are measured in pixels from the lower left of the graphics area. The Processor determines which character cells in the graphics area are needed to contain the boundaries of the figure. It then figures out which pixels need to be turned on within these character cells. As it is doing this, the Processor finds unused locations within Graphics Memory and stores character cells with the appropriate pixel data there.

   The Processor inserts the fond and character addresses for these character cells within the display list blocks reserved for the graphics area. The character address in bytes 4 and 5 of block 6 selects the pixel data for the left top portion of the triangle. Inside the triangle, there is one character cell that is solid C5. Notice that in the display list (bytes 7 and 8 of block 7), the address for this is font 31, character 127. when the terminal is initialized, the Processor stores a filled-in character cell for each of the eight color numbers at the eight highest addresses in Graphics Memory. Then, when solid colors are needed for filling in areas, the Processor has only to call these characters. It doesn't have to store data in graphics memory for each cell inside the figure.

D. A workspace has been defined. This divides the screen into two areas: workspace and monitor. These two areas on the screen can be thought of as two windows through which two separate display lists can be viewed. Each of these "separate" display lists can be scrolled up and down in its respective window. The display list is divided into monitor and workspace lists by Tracker jump commands in the firt 16-byte block of the display list.

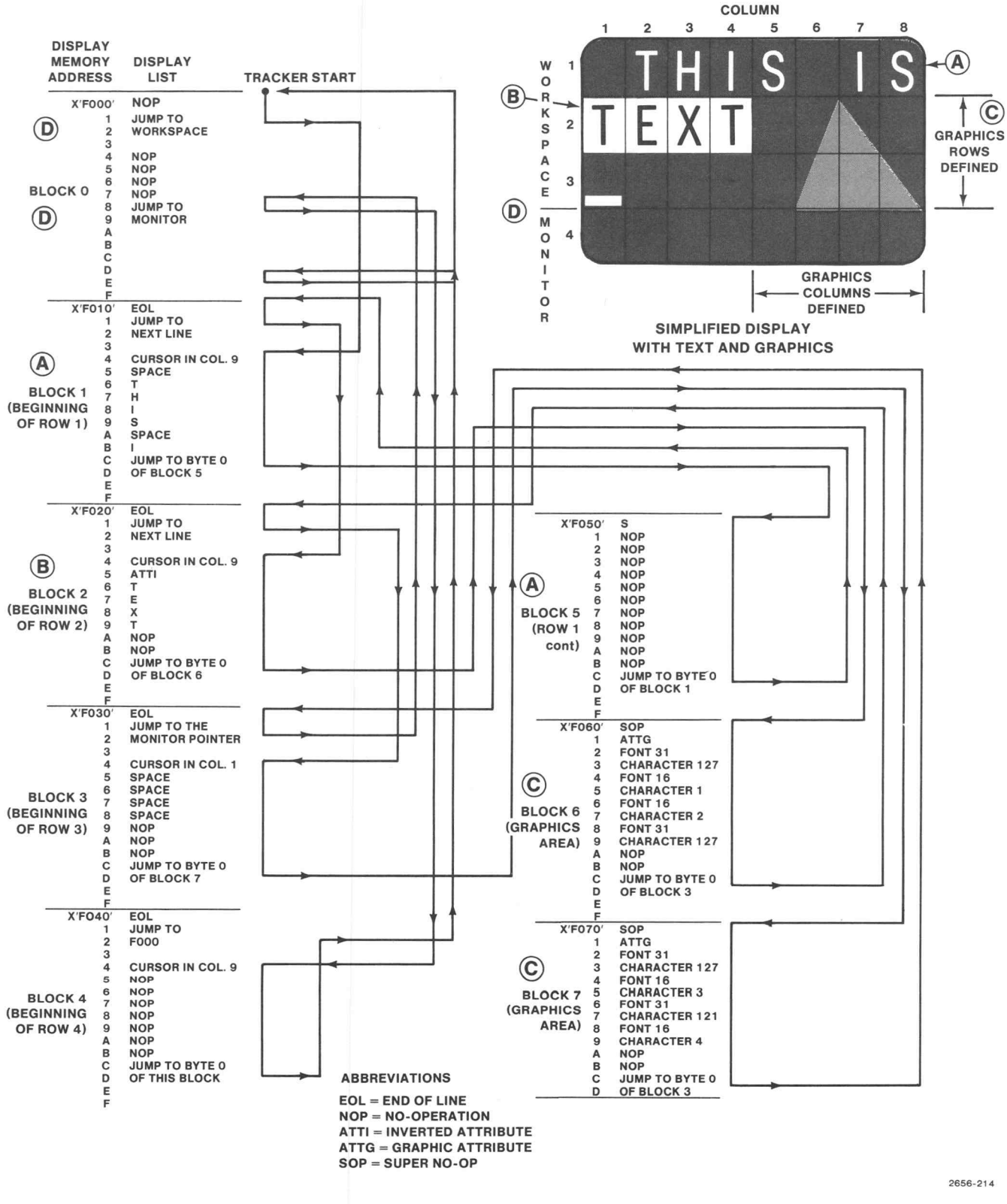


Figure 3-3. Simplified Display with Text and Graphics Added.

The Tracker starts by jumping to the first block of the first character row currently in view in the workspace part of the display list. It goes through all of the blocks for all of the character rows in view in the workspace. From the last block in the workspace, the Tracker jumps back to bytes 8 and 9 of the pointer block. These bytes contain a jump instruction which sends the Tracker to the first block in the monitor. It goes through the monitor, then jumps back to the pointer block where it remains in a loop at D and E until it is reset to X'F000' after vertical retrace.

To scroll the display, the Processor simply modifies the necessary jump addresses in the pointer block and in the body of the display list, so that the Tracker is routed through a new portion of the display list.

Section 4

DETAILED CIRCUIT DESCRIPTIONS

This section contains detailed, comprehensive information on the 4027A's circuitry. In order to use this information, it may be necessary to understand certain basic concepts presented in earlier sections.

There is a detailed block diagram for almost every one of the circuit boards. In general, each block on these diagrams corresponds to a paragraph or subsection in the text and to an area outlined by gray, dashed lines on the schematic diagrams (schematics are contained in Vol. 2 of this service manual). For example, for the Color Map Circuit on the Display Controller Board, there is a block labeled COLOR MAP on the Display Controller Block Diagram, Figure 5-3. Likewise, on Schematic 2-4, there is an area set off by gray dashed lines and labeled COLOR MAP. In the text, under Display Controller, there is a subsection titled "Color Map" which contains a description of the Color Map circuit.

Three overall block diagrams, Figures 5-1, 5-2, and 5-10, when used with the detailed block diagrams, may be helpful in understanding how a certain circuit fits into the terminal's overall functional organization.

DISPLAY MEMORY BOARD

The Display Memory occupies the upper half of the microprocessor's memory address space. (If 16K of display memory is installed, it occupies addresses X'C000' to X'FFFF'. With the maximum 32K of memory installed, it occupies X'8000' to X'FFFF'. It serves as a general-purpose "pool" of read/write memory. For instance, it is in Display Memory that the Processor stores input/output queues and the definitions of the programmed functions keys. However, the most important thing stored in Display Memory is the display list.

Refer to Figure 5-3. The Display Memory Board includes these major circuit blocks:

- Display RAM
- RAM Controller
- RAM Address Steering
- Tracker
- Bus Interface

Display RAM

The Display memory consists of either eight or sixteen RAM devices, each holding 16K bits, as follows:

- 16K Standard: eight 16Kx1 devices providing 16,384 bytes.
- 32K (Option 22): sixteen 16Kx1 devices providing 32,768 bytes.

These RAMs appear in Schematic 1-3; Figure 4-1 shows their pinout. The 16K RAMs have seven address lines which alternate between two halves of a 14-bit address. The **row address** is the first half of the address placed on a RAM's address pins; the **column address** is the second half to be placed on those pins.

Figure 4-2 shows the RAM read, write, and refresh timing.

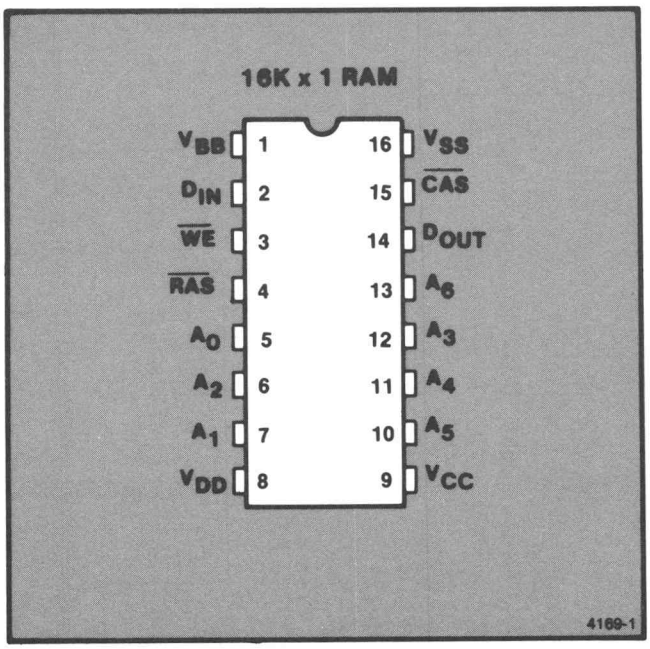


Figure 4-1. Pinout for 16K RAMs.

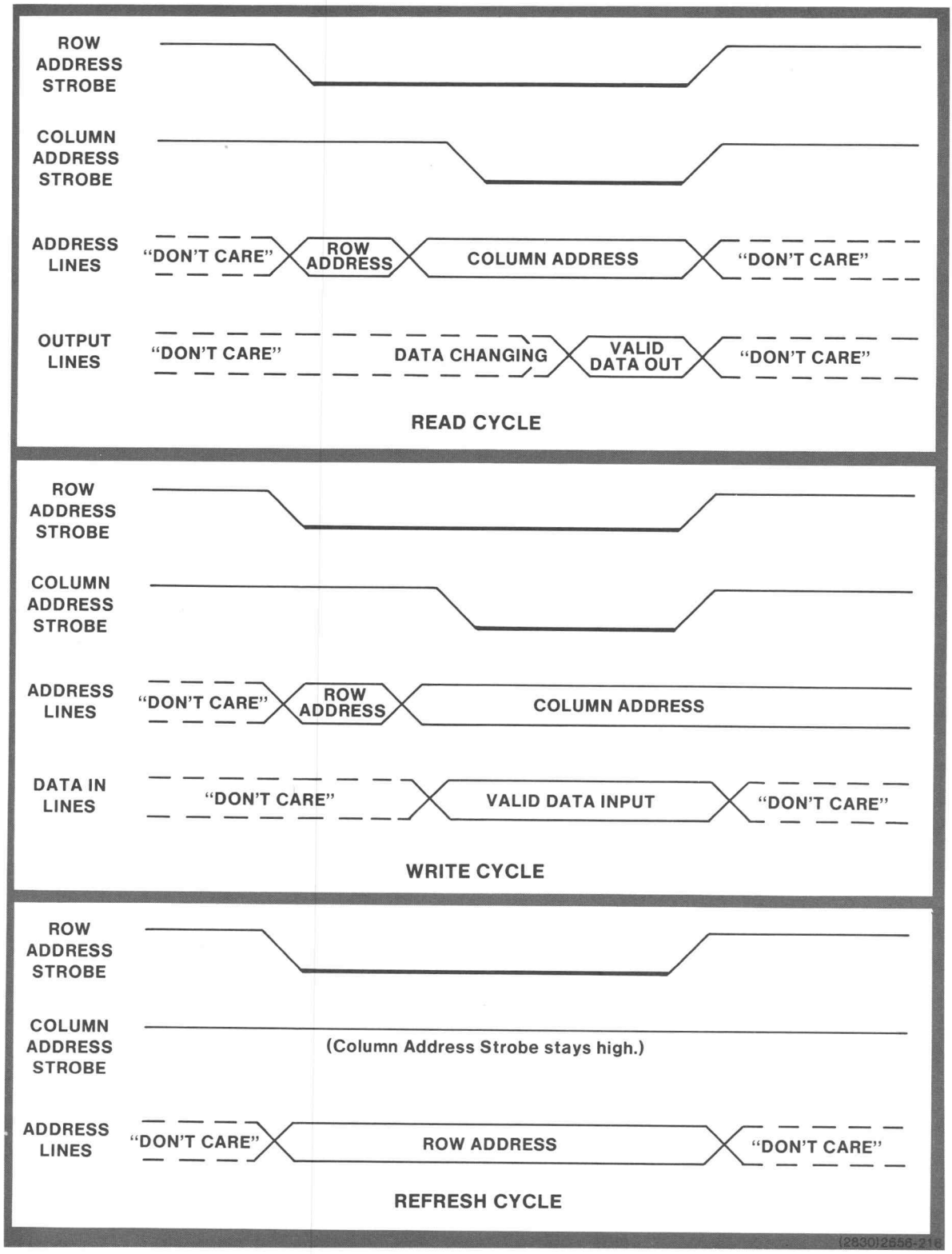


Figure 4-2. RAM Waveforms.

RAM Controller

(Figure 5-3, Schematic 1-1)

The RAM Controller is divided into three parts: the Cycle Type Pointer, the RAM Control State Machine, and the Refresh Request Counter. It provides signals to control the RAMs during read, write, or refresh cycles. It also provides periodic read cycles to refresh the dynamic RAMs. During a read or write cycle, it provides the correct RAS (Row Address Strobe) and CAS (Column Address Strobe) signals to control the multiplexing of the RAM address inputs.

The RAM Controller inputs are:

- HCLK. The 18.432 MHz clock.
- LCLK. The 2.0428 MHz clock.
- BCRQ (Bus Cycle Request). A signal that the Processor requests a memory read or write operation.
- TCRQ (Tracker Cycle Request). A signal that the Tracker requests a read operation.
- JUMPING. A Tracker signal indicating that the byte being read is the second byte of a jump command and should be loaded directly into the Tracker Address Counter rather than into the Tracker Data Latch.

The RAM Controller outputs are:

- RAS (Row Address Strobe). A signal asserted whenever a RAM is to read its address inputs.
- CAS (Column Address Strobe). A signal asserted whenever the second half of the RAM address (the column address) has settled, and the RAM is to read that column address from its address inputs.
- RAD (Refresh Addressed). A signal that the RAM Controller's Cycle Request Pointer is pointing to "refresh cycle." During a refresh cycle, this signal causes the RAM address to be taken from a "refresh address counter" rather than from the Tracker or the terminal address bus.
- BAD (Bus Addressed). A signal indicating that the Cycle Request Pointer is pointing to a "bus cycle." This causes the RAM address to be taken from the terminal's main address bus.

Cycle Type Pointer

(Schematic 1-1)

The Cycle Type Pointer consists of the flip-flops labeled P0 and P1 in Schematic 1-1. These comprise a two-bit binary counter which repeatedly counts from 0 to 3. The counter "points" to the type of RAM cycle which may occur at any instant. On the counts of 0 and 2, the pointer designates a **bus cycle** — a RAM cycle requested by the Processor. On the count of 1, it points to a **Tracker Cycle** — a RAM cycle requested by the Tracker. On the count of 3 it designates a **Refresh Cycle** — a RAM read cycle requested by the Refresh Request Counter.

The P0 and P1 flip-flops control data selector U115, which selects whether the RAM Control State machine takes its cycle request inputs from Refresh Request Counter U311, from BCRQ (Bus Cycle Request) line, or from the TCRQ (Tracker Cycle Request) line.

The P0 and P1 flip-flop outputs are also decoded to generate the RAD (Refresh Cycle Addressed) and BAD (Bus Cycle Addressed) signals.

RAM Control State Machine

(Schematic 1-1)

The RAM Control State Machine consists of the flip-flops labeled Q0 to Q4 and their associated gates. The flip-flop outputs make up a code designating which state the State Machine occupies, as follows:

STATE	Q4	Q3	Q2	Q1	Q0
A	0	0	0	0	0
B	0	0	0	0	1
C	0	0	0	1	1
D	0	0	1	1	1
E	0	1	1	1	1
F	1	1	1	1	1
G	1	1	1	1	0
H	1	1	1	0	0
I	1	1	0	0	0
J	1	0	0	0	0

As the State Machine moves through these states, it generates signals which drive the RAMs in a read or write cycle. The states are as follows:

State A: Advance the Cycle Type Pointer. The next state is B.

State B: Ask, "Is a RAM cycle being requested?" If "Yes," the next state is C. If "No," the next state is A.

(If no device requests a RAM cycle, the State Machine alternates between states A and B. When a device requests a cycle, the State Machine waits until the Pointer designates the requesting device, then initiates the RAM cycle by entering state C.)

State C: Assert RAS (Row Address Strobe) and place the row address on the RAM address bus. (The row address is the first half of the RAM address to be placed on the RAM address bus.) The next state is D.

State D: Continue as in state C. The next state is E.

State E: Continue asserting RAS. On the RAM address bus, replace the row address with the column address. The next state is F.

State F: Continue asserting RAS and holding the column address on the RAM address bus. By now the column address has settled; assert CAS (Column Address Strobe). The next state is G.

State G: Continue sending RAS and CAS and holding the column address on the RAM address bus. The next state is H.

State H: Continue as in states F and G. The next state is I.

State I: Continue asserting RAS and CAS. By now the RAM output data should have settled; latch that data. If this is a "Bus Cycle," latch the data in the Bus Latch for later transfer to the Processor over the main data bus (send the BLTH signal.) If this is a "Tracker Cycle," put the data in the Tracker Latch (send the TLTH signal) unless the Tracker is on the second half of a "jump" (JUMPING true); if this is the case, put the data in the Tracker Address Counter (send the LCNT signal).

The next state is J.

State J: Turn off any signals being sent. This concludes the RAM cycle.

The next state is A.

The gates scattered throughout the RAM Control State Machine decode its states, providing the output signals and the flip-flop inputs to advance from one state to the next.

Refresh Request Counter

(Schematic 1-1)

The Refresh Request Counter in the RAM Controller is driven by LCLK, the terminal's 2.048 MHz clock. It periodically sends a signal to data selector U115, indicating that it's time for another RAM refresh cycle. Setting jumper SB to position 16 allows 16K RAMs (if installed) to be refreshed more frequently than 4K RAMs would be.

RAM Address Steering

The RAM Address Steering circuitry interfaces the RAM address lines with the address sources to be placed on those lines. This interface is needed for two reasons:

- There are three possible sources for an address to be placed on the RAM address bus: the Processor Address Bus, the Tracker Address Counter, and the Refresh Address Counter. There must be a way to switch the RAM address bus between these three sources of address information.

- As mentioned before, the RAM addresses are multiplexed; only seven RAM address lines are used to carry 14 address bits. There must be a way to switch the RAM address lines between the seven "row address" bits and the seven "column address" bits.

The RAM Address Steering circuitry (Figure 5-3, Schematic 1-2) consists of the following blocks: RAM Address Selectors, Address Multiplexer & Refresh Counter, and RAM Enable Gates.

RAM Address Selectors

Several "data selector" devices (U471, U481, U375, U365, U475) select whether a RAM address comes from the Tracker Address Counter or the main terminal bus. These data selectors are steered by the BAD (Bus Cycle Addressed) signal from the RAM Controller. They pass 14 address bits on to U485.

Address Multiplexer and Refresh Address Counter

U485, an Intel 3242 (or equivalent), is a combination RAM address multiplexer and refresh address counter. When the RAD (Refresh Cycle Addressed) signal is false, U485 places either the row address bits or the column address bits (depending on the state of the ROW signal) on the RAM address bus.

When RAD and ROW are both asserted, U485 places the row address of the row to be refreshed on the RAM address bus. The row address is provided by a Refresh Address Counter (within U485), which is advanced on each refresh cycle.

RAM Enable Gates

Several gates associated with the RAM Address Steering circuitry provide signals to enable the two banks of RAMs:

- CS drives the RAM column address strobe pins whenever CAS is asserted as it is not a refresh cycle (RAD not true).
- RL (RAS for the "low" bank of RAMs) drives the row address strobe pins of the "low" bank of RAMs whenever the RAM Controller is addressing one of those RAMs.
- RH serves a similar function for the "high" bank of RAMs.
- WR (Write) drives the RAM write enable pins during a write cycle initiated by the Processor (a "bus write cycle").

Tracker

(Figure 5-3, Schematics 1-1, 1-2, and 1-4)

The Tracker circuit follows the display list through Display memory, operating on "jumps," "no-ops," "end-of-line markers," and other Tracker commands contained in the display list. It passes the character and font addresses and visual attribute codes to the Display Controller. The Tracker's circuitry consists of the Tracker Data Latch, Tracker Input Decoder, Blink Counter, Tracker State Machine, Tracker Reset Circuit, and Tracker Address Counter.

Display List Contents

Figure 4-3 depicts the formats of the various kinds of information contained in the display list.

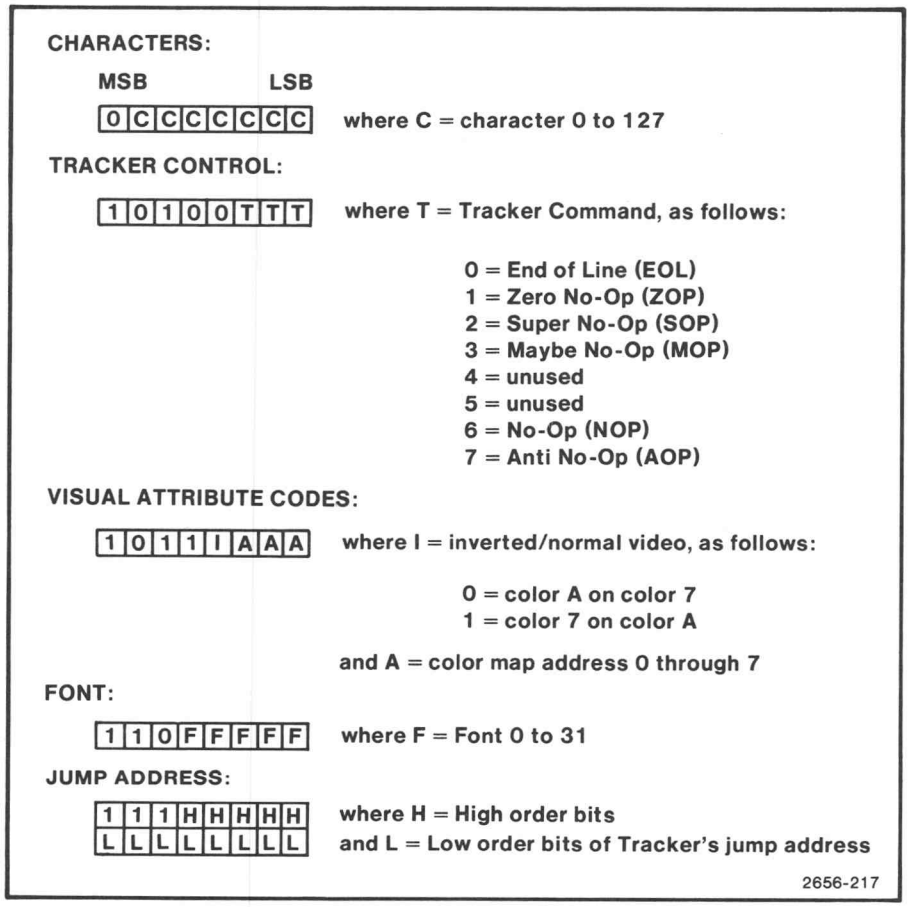


Figure 4-3. Display List Contents.

Character and Font Addresses. The most common items in the display list are the Character Memory addresses for the text and graphics characters in the display. Each character address specifies one of 128 dot patterns within a font. For example, font zero contains 120 different dot patterns for the ACII characters (including the control characters).

Attribute Codes. There are two different kinds of attribute codes: visual attribute codes and logical attribute codes. The Tracker passes the visual attribute codes on to the Display Controller.

Logical attribute codes are inserted in the display list only for the Processor's benefit; they do not affect how text is displayed. Because of this, the Processor inserts, before each logical attribute code, a command ("super no op") for the Tracker to skip over that code.

Tracker Control Commands. The Tracker Control commands are: EOL, NOP, ZOP, SOP, MOP, and JUMP.

EOL (End of Line): EOL marks the end of a row of text. When the Tracker encounters an EOL, it briefly pulls the BUFUL (Buffer Full) line low, signaling the Display Controller to reset the font address and visual attribute to font zero and "standard" visual attribute, respectively. The Tracker then repeatedly sends ASCII spaces (character number 32) to the Display Controller. This continues until the Display Controller reaches the end of its Row Buffer. The Display Controller then pulls BUFUL low and holds it low. This causes the Tracker to wait. When BUFUL goes high again, the Tracker starts assembling the next row of text.

NOP (No Op): Upon reading a NOP, the Tracker goes on to the next byte (word of memory) without doing any other operation.

ZOP (Zero No Op): When the Tracker reads a ZOP, it reads the same word over and over, until it finds there a command other than ZOP. ZOP is used during modification of JUMP commands to keep the Tracker from reading a partially modified JUMP, because the JUMP is two bytes long. In order to modify a JUMP, the Processor first changes the first byte to a ZOP. It then changes the second byte to the second byte of the new JUMP command. Finally, it changes the ZOP to the first byte of the new JUMP command.

SOP (Super No Op): SOP tells the Tracker to ignore (skip over) both the SOP and the next word in the display list.

MOP (Maybe No Op): The MOP instruction is used to blank visual attribute fields. There is a flag (BLNKFLAG) which changes levels every quarter second. When the flag is high, Tracker decodes a MOP as a SOP: it skips over both the MOP and the following word of the display list. When the flag is low, Tracker decodes a MOP as NOP: it skips over only the MOP.

AOP (Anti No Op): This is the opposite of MOP; it functions like a NOP when BLNKFLAG is high, and like a SOP when U411, Pin 11 is low.

JUMP: This is a two-byte command which causes the Tracker to jump to another location in the display list.

Tracker Latch and Input Decoder

(Schematic 1-4)

Data read from RAM during a Tracker RAM cycle is latched in U351 (the Tracker Latch) and becomes the display list byte on which the Tracker operates. The Tracker Input Decoding gates examine this word and provide the Tracker State Machine with the following signals:

- EOL: The word just read from the display list is an end-of-line marker.
- ZOP: The word just read from the display list is a "zero no op" command.
- SOP: The word just read from the display list is a "super no op" command.
- SPEC: The word just read from the display list is a "special Tracker command": EOL (End-of-Line), ZOP (Zero No Op), SOP (Super No Op), MOP (Maybe No Op), NOP (No Op), or the first half of a JUMP.
- JUMP: The word just read from the display list is the first half of a two-byte JUMP command.

Blink Counter

The Blink Counter (U411, Schematic 1-4) is clocked by VDRIVE pulses from the Display Controller. On every 128 VDRIVE pulse, its Blink Flag (U411, Pin 11) output toggles an exclusive-OR gate in the Tracker Input Decoding circuitry.

As the Blink Counter toggles, the Tracker Input Decoding alternates in how it interprets MOP (maybe no op) and AOP (anti no op) instructions. MOP is used for "blinking" between two sets of visual attributes.

Tracker State Machine

(Schematic 1-1)

The heart of the Tracker is the Tracker State Machine. This is a specialized processor which acts on the signals from the Tracker Input Decoders and the RAM Controller, and provides signals to:

- Request Tracker cycles from the RAM Controller (the TCRQ signal).
- Inform the RAM controller when the word to be read is the second half of a Tracker JUMP command (JUMPING).
- Load the high order bits of the Tracker Address Counter (send the LHCNT signal). (The low-order bits are loaded by the RAM Controller's LCNT signal.)
- Inform the Display Controller when to clock another word into its input buffer (TCLK) and when that buffer is full (BUFUL). (The BUFUL signal makes the Display Controller reset the current visual field attribute to "standard" and the character font to "font 0."

Latch U21 and ROM U15 are the heart of the state machine. LCLK, the terminal low frequency clock, clocks the latch to advance the state machine from one state to the next. The latch outputs provide address inputs for the 32 x 8 ROM. Three of the ROM inputs (U15 Pins 10, 11, and 12) give the "present state address." The other two ROM inputs (Pins 13, 14) are the data inputs for that state. Three of the ROM output bits (Pins 1, 2, 3) specify the next state. Two output bits (Pins 7, 8) control data selector U111, selecting which two signals will be examined as inputs in the next state. The remaining three ROM output bits (Pins 4, 5, 6) determine which signals the State Machine sends to the RAM Controller, Display Controller, etc.

ROM address input B (Pin 11) serves a special purpose: it is high when the next Tracker RAM cycle will be reading the second byte of a jump command. This is passed as a signal (JUMPING) to the RAM Controller so that the RAM controller loads the byte into the Tracker Address Counter, rather than into the Tracker Data Latch.

Demultiplexer U315 decodes the ROM output bits (ROM outputs Y6, Y5, Y4) and, together with some associated gates and flip-flops, generates the Tracker output signals:

- When ROM outputs Y6, Y5, Y4 are 0, 0, 0, the demultiplexer clears the TCRQ flip-flop, U215A. The flip-flop then sends a TCRQ (Tracker Cycle Request) signal to the RAM Controller. (The flip-flop is reset by a TCYCDUN, Tracker Cycle Done, signal from the RAM Controller.)

- When ROM outputs Y6, Y5, Y4 are 0, 0, 1, an SCNT signal is sent (to advance the Tracker Address Counter), provided the byte in the Tracker Data Latch is not a ZOP command.

- When ROM outputs Y6, Y5, Y4 are 0, 1, 0, the Tracker Address Counter is advanced if the byte in the Tracker Data Latch is an SOP command.

- When ROM outputs Y6, Y5, Y4 are 0, 1, 1, the Tracker Address counter is advanced, unconditionally. Also, the TCLK (Tracker Clock) signal is sent to the Display Controller; the Display Controller then accepts the byte waiting for it on the paired bus lines (outputs of the Tracker Latch).

- When ROM outputs Y6, Y5, Y4 are 1, 0, 0, the BUFUL signal is sent to the Display Controller.

- When ROM outputs Y6, Y5, Y4, are 1, 1, 0, the Tracker State Machine sends no signals, but just advances to its next state.

- When ROM outputs Y6, Y5, Y4, are 1, 1, 1, the Tracker sends a TCLK (Tracker Clock) signal to the Display Controller. (The Display Controller then seizes the byte presented it on line T0-T7, the outputs of the Tracker Latch.)

Tracker Reset Circuitry

(Schematic 1-4)

The Tracker Reset Circuitry provides the RCNT (Reset Counter) signal, which resets the Tracker to start again at the first line of text to be displayed. RCNT loads X'F000' into the Tracker Address Counter. (This address holds a JUMP directing the Tracker to the start of the first character row to be displayed.) RCNT also clears the state address latch in the Tracker State Machine; this re-initializes the state machine to a known starting state.

The Tracker Reset Circuitry has three inputs, corresponding to three events which can reset the Tracker to X'F000':

- The Mother Board RESET line provides a signal on power-up or when the MASTER RESET button is pressed.
- The paired bus VDRIVE line provides a signal at each vertical retrace.
- The paired bus HC-0 line provides a signal when a hard copy is in progress.

A three-input negative-logic OR gate (U131C) detects when any of the three conditions occurs. This gate's output is ANDed with the CLK signal to provide the RCNT signal.

Since the 4027A uses the 4632 Video Hard Copy Unit, the HCU (Hard Copy Unit) jumper is set to VID (Video). With the jumper in this position, AND gate U421D always passes the VDRIVE-1 signals to U131C.

Tracker Address Counter

(Schematic 1-2)

The Tracker Address Counter holds the address of the next byte which the Tracker will read from the display list. The Tracker advances this counter to go from one byte in the display list to the next.

When the Tracker encounters the first bytes of a jump command, it leaves that byte stored in the Tracker Latch (Schematic 1-1). It then requests another RAM cycle, sending the JUMPING signal as it does so. JUMPING tells the RAM Controller to take the byte it reads from RAM and load it into the low-order bits of the Tracker Address Counter. (Usually, a byte read during a Tracker Cycle is loaded into the Tracker Latch; JUMPING causes it to go into the Tracker Address Counter instead.)

At the end of the Tracker Cycle, TCYCDUN goes high. This clocks the TCRQ flip-flop (Schematic 1-1), turning off the TCRQ signal. Also, if JUMPING is true, the positive edge of TCYCDUN clocks flip-flop U215B in the Tracker. This flip-flop then sends the LHCNT signal, which loads bits stored in the Tracker Latch into the high-order bits of the Tracker Address Counter.

Bus Interface Circuitry

(Figure 5-3, Schematics 1-2 and 1-4)

The Bus Interface circuitry interfaces the Display Controller Board to the main terminal bus. It includes: RAM Address Decoder, Bus Data Latch, Bus Read Buffers, Bus Write Buffers, I/O Address Decoder, Status Register, and Interrupt Request Circuitry.

RAM Address Decoder

(Schematic 1-2)

The RAM Address Decoder decodes the high-order bits of the terminal data bus to see whether display memory is being addressed. If it is, and either WRITE-0 or READ-0 is asserted, it sets the Bus Cycle Request flip-flop and the Wait Latch, which perform a "handshake" function.

As these flip-flops are set, a BCRQ (Bus Cycle Request) signal goes to the RAM Controller, and a WAIT signal goes to the Processor. When the requested RAM cycle is complete, BLTH (which latches the data into the Bus Data Latch) resets the flip-flops.

Bus Write Buffers

(Schematic 1-4)

The Bus Write Buffers are inverters which buffer the Mother Board data lines BD0-BD7, driving the on-board data lines E0-E7.

Bus Data Latch and Bus Read Buffers

(Schematic 1-4)

During a Processor-initiated RAM cycle (bus cycle), the data read from the RAMs is latched into the Bus Data Latch by the BLTH signal from the RAM Controller. The DENAB (Data Enable) signal (from the Bus Interface circuitry on Schematic 1-2) causes the Read Buffer to place this data on the main terminal bus. This DENAB signal occurs when (a) the Display Memory Board has been addressed, as detected by the Board Address Decoder, and, moreover, (b) the Processor has sent a READ signal on the main terminal bus.

I/O Address Decoder

(Schematic 1-4)

A network of gates detects reads or writes to addresses X'0810' or X'0811'. (The gates do not decode all the address lines: they treat any even address in the range X'0810' to X'081F' the same as X'0810'; likewise, any odd address in that range is regarded as address X'0811'.) Address X'0810' holds the Status Register, while X'0811' is used for clearing VDRIVE interrupt requests.

Table 4-1 lists the Display Memory Board I/O Register functions.

Table 4-1

DISPLAY MEMORY BOARD I/O REGISTERS

Address	Register	Functions
X'0810'	Status Register	Write: Bit 0: INTEN (INTEN=1: enable interrupts) Bit 1: Unused Bit 2: HC (To start a hard copy, set HC to 1.) Bits 3-7: Unused
		Read: Bit 0: INTEN Bit 1: VDRIVE detected Bit 2: HC (HC=1; hard copy in progress) Bits 3-6: Unused Bit 7: VDRIVE interrupt being requested
X'0811'	VDRIVE Detector Reset	Write: Writing into X'0811' clears the VDRIVE Detect Flip-Flop.
		Read: (The "read" half of address X'0811' is unused.)

Status Register

(Schematic 1-4)

The Status Register occupies address X'0810' (and all other even addresses between X'0810' and X'081F'). By writing into this register, the Processor can enable interrupts from the Display Memory Board or initiate a hard copy. By reading from this register, the Processor can learn:

- Whether "VDRIVE detected" interrupts are enabled.
- Whether a VDRIVE pulse has occurred.
- Whether a "VDRIVE detected" interrupt has occurred.
- Whether a hard copy operation is in progress.

The Status Register includes: VDRIVE Detect Flip-Flop, IRQ Enable Flip-Flop, Status Register Read Buffer, and Hard Copy Start circuitry.

VDRIVE Detect Flip-Flop: VDRIVE pulses from the Display Controller occur 60 times per second; they are used as a timer by the system firmware. Each pulse sets the VDRIVE Detect Flip-Flop. The flip-flop is cleared by a write to address X'0811'.

IRQ Enable Flip-Flop: The Processor enables VDRIVE interrupts by writing a one into bit 0 of the Status Register. This clears a type D flip-flop; when the flip-flop is cleared, VDRIVE interrupts are enabled. Writing a zero into the same bit sets the flip-flop, disabling the interrupts.

Status Register Read Buffer: Four tri-state buffers are enabled by a read from address X'0810'. When enabled, they place the following information on four bits of the Processor's data bus:

- Bit 0 tells whether VDRIVE interrupts are enabled.
- Bit 1 tells whether a VDRIVE pulse has occurred, that is, whether the VDRIVE Detect Flip-Flop is set.
- Bit 2 tells whether a hard copy is in progress, that is, whether the HC line on the Mother Board is low.
- Bit 7 is true (a logical 1) if VDRIVE interrupts are enabled and a VDRIVE pulse has been detected.

Hard Copy Start: Three gates (two of them functioning as Schmitt trigger inverters) detect when the Processor writes a 1 into bit 2 of the Status Register. When this occurs, the HC (Hard Copy) line briefly goes low. This starts the hard copy. Circuitry on the Display Controller Board and the 4632 Hard Copy Unit will hold HC low until the hard copy operation is complete.

Interrupt Request Circuitry

(Schematic 1-4)

The Interrupt Request Circuitry consists of three gates and a jumper. In the 4027A the jumper is set to VEC (for "vectored interrupts"). With the jumper in this position, the gates drive the IRQ line low only when a VDRIVE signal has been detected (with VDRIVE interrupts enabled) and the interrupt address lines IA0-IA2 hold the Display Memory Board's interrupt address (binary 010).

DISPLAY CONTROLLER BOARD

The Block Diagram for the Display Controller Board is Figure 5-4. All the digital information for creating the display comes together in the Display Controller. It receives character addresses and visual attribute codes from the Display Memory Board and stores these in its Row Buffers. It then uses the character addresses to retrieve the appropriate pixel data from the character memories. The pixel data is loaded into shift registers and converted to a serial video data stream. This video data is sent, along with horizontal and vertical sync signals to the Digital to Analog Converters on the Z-Axis and Convergence Board.

Timing Generator

Refer to Schematic 2-1 and Figure 4-4. The purpose of the Timing generator is to synchronize all functions on the Display Controller Board and many functions in other areas of the terminal. A video sync generator integrated circuit is central to the Timing Generator circuit. This device divides down the processor-generated 2.048 MHz clock, LCLK-1, to produce several timing signals.

The outputs of the video sync generator are loaded into a latch on each positive transition of LCLK-1. This causes all the timing signal transitions to be synchronized to LCLK-1 (and therefore to the Processor).

The outputs of the latch are:

- VDRIVE-0 (Vertical Drive). This signal goes true once every 16.7 ms, at the beginning of vertical retrace time. There is one VDRIVE-0 for each raster field. VDRIVE-0 lasts for 9 HDRIVE-0 periods.

- HDRIVE-0 (Horizontal Drive). This signal occurs once each horizontal scan and it is the basic timing signal for events related to this period (63.5 μs, approximately). It comes at the beginning of horizontal retrace and is true for approximately 6.35 μs.

- CBG-0 (Color Burst Gate). CBG-0 occurs at the same rate as the Horizontal Drive signal. It goes true one LCLK-1 period after the end of HDRIVE-0 and remains true for 5 LCLK-1 periods. CBG-0 ceases to occur during the time when VDRIVE-0 is true.

- CBO-0 (Composite Blanking Output). CBO-0 is used by the Video Output circuit to generate the BLANK output (for external monitors). It also is used within the Timing Generator to produce additional timing signals.

- SYNC-0 (Composite Sync Output). This is used by the Video Output circuit to generate SYNC (for external monitors) and HSYNC-0 (for the horizontal deflection circuits).

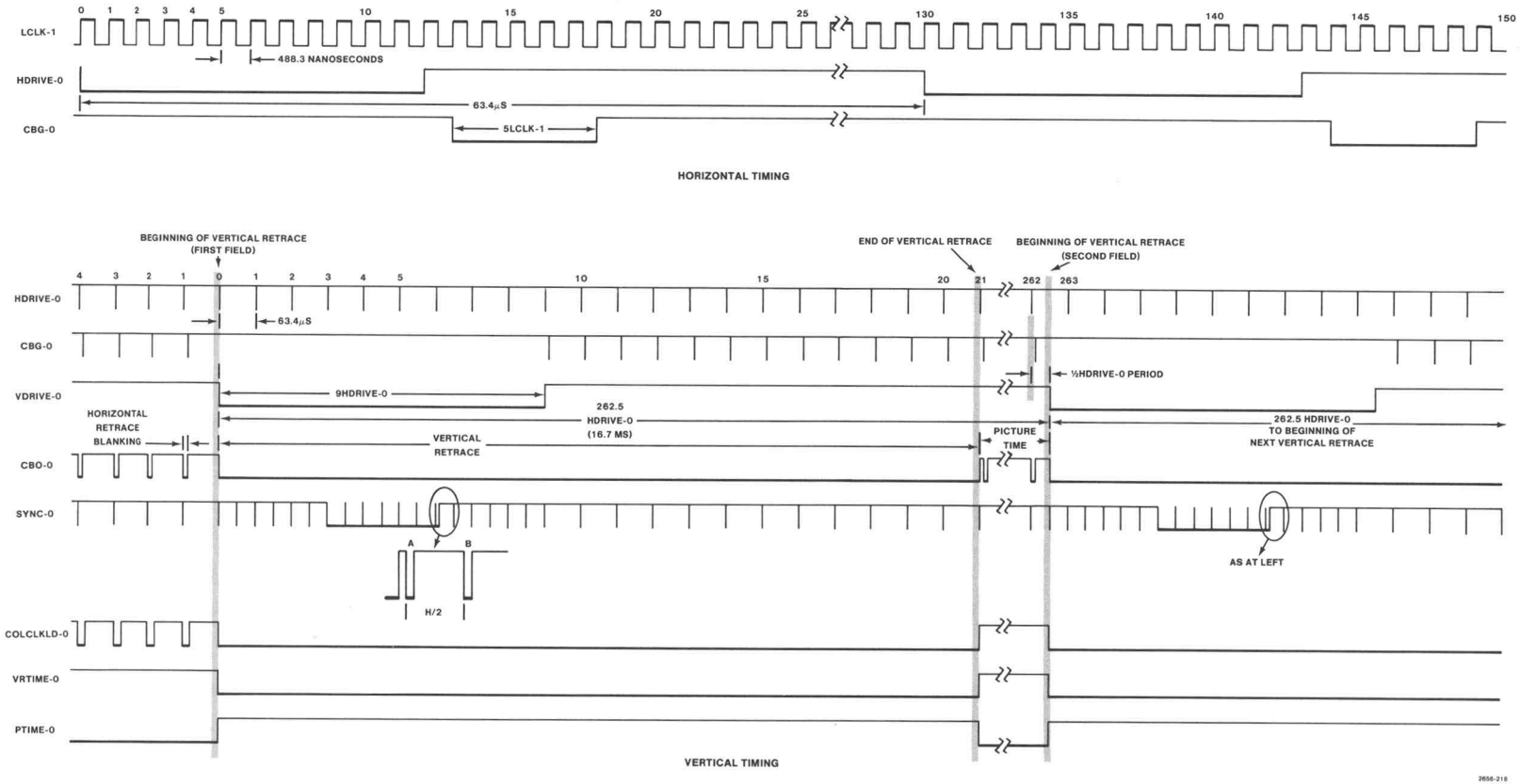


Figure 4-4. Horizontal and Vertical Timing.

The foregoing signals are used to produce several additional timing signals within the Timing Generator. They are:

- PTIME-0 and PTIME-1 (Picture Time). These are true during the interval between the first positive transition of CBG-0 after vertical retrace time ends and the beginning of the next VDRIVE-0.

- COLCLKLD-0 and COLCLKEN-0 (Column Clock Load and Enable). These signals are complementary. COLCLKLD-0 loads the Column Clock Counter during retrace times. COLCLKEN-0 allows the counter to count during scan periods and disables it during retrace.

- VRTIME-0 and VRTIME-1 (Vertical Retrace Time). These complementary signals are true from the beginning of Vertical Drive until the first positive transition of CBO-0 after VDRIVE-0 (approximately 21 Horizontal periods).

- SCANCNT-1 (Scan Count). This signal occurs in step with CBG-0 (i.e., once every horizontal scan). When VDRIVE-0 is true, SCANCNT-1 remains in a low state.

Bit Clock

Refer to Schematic 2-1 and Figure 5-4. HCLK-1, the 18.432 MHz clock from the Processor is divided by 1.5 in the Bit Clock circuit to produce BITCLK-1 and BITCLK-0. One BITCLK period is the time required to produce one pixel on the display screen (81.4 ns). BITCLK-0 is used to shift the data out of Shift Registers A, B, and C and to clock the X Cursor Counter and the Column Clock Counter. BITCLK-1 Strobes the output of the Color Map.

In order to synchronize BITCLK to the horizontal scan, the Bit Clock circuit is gated off and on by CBG-0. When CBG-0 goes true, it holds the preset inputs to two flip-flops low, thereby turning off BITCLK. BITCLK starts up again when CBG-0 goes high. This ensures that BITCLK starts at the same time on each horizontal scan.

Column Clock

Refer to Schematic 2-1 and Figure 5-4. The Column Clock circuit consists of a modulo-16 counter, a NAND gate and an inverter. This circuit produces COLCLK-1, BITSYNC-0, and COLSYNC-0.

The counter is clocked by BITCLK-0. The period of COLCLK-1, which is taken from the Qc output of the counter, is 8 BITCLK-1 cycles long. BITSYNC-0 is produced by a 3-input NAND gate connected to the high-order counter outputs. It goes true on every eighth count of the counter and remains low for one BITCLK-0 period.

The periods of both COLCLK-1 and BITSYNC-0 are one character cell wide in time (650 ns) and these signals are used to control events related to this period. For example, a Display Controller access to Graphics Memory is synchronized to COLCLK-1 and BITSYNC-0 latches the data returning from Character Memory into the Shift Registers A, B, and C.

COLSYNC-0 is the ripple carry output of the Column Clock counter. It goes true once every 16 BITCLK-0 cycles. Therefore, its period is two "character cells" long. COLSYNC-0 is used in the Buffer Control and Blanking Flip-Flop circuits where a two character delay is needed.

Sync Output

Refer to Schematic 2-1 and Figures 4-5 and 4-6. The Sync Output circuit receives five inputs from the Timing Generator. It processes these inputs to produce several synchronizing signals. The following list summarizes the characteristics of these outputs.

- BLANK. This output is a level shifted version of CBO-0. It is intended to be used in external video systems for blanking control.
- SYNC. This is a level shifted version of SYNC-0 which can be used to synchronize external video systems.
- VSYNC-1 (Vertical Sync). This is a 200 μs pulse triggered on the positive edge of VDRIVE-0. It synchronizes the vertical deflection circuits.
- VDRIVE-0, HDRIVE-0, CBG-0, and CBG-1. These are buffered versions of the same signals described under the Timing Generator subsection.
- SYNC-1. This is SYNC-0, inverted.
- HSYNC-1. This is the inverted, logical AND of the HDRIVE-0 and SYNC-0 signals. It occurs in step with HDRIVE-0 but changes slightly in phase and duration during vertical retrace periods.

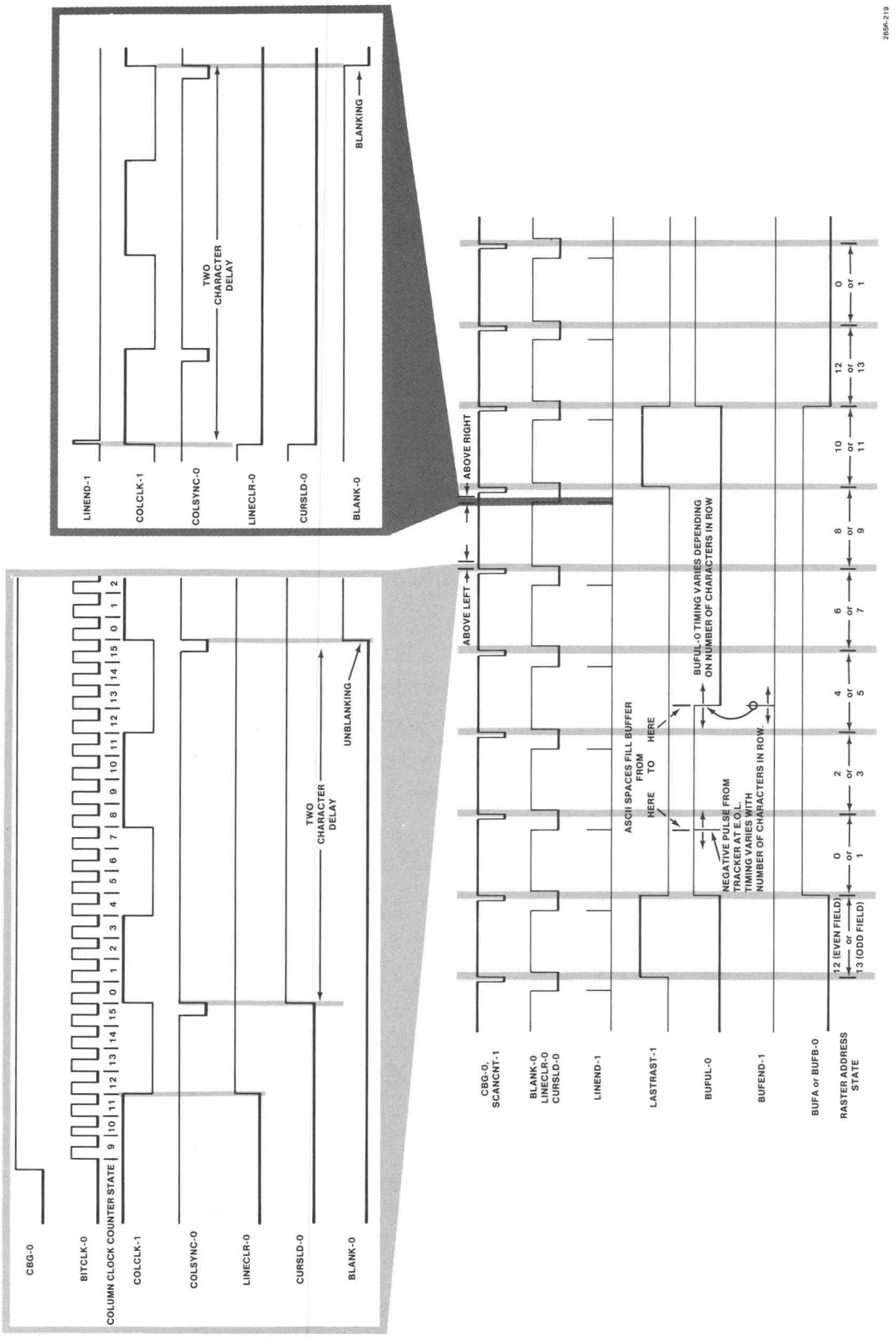


Figure 4-5. Buffer Control and Column Clock Timing.

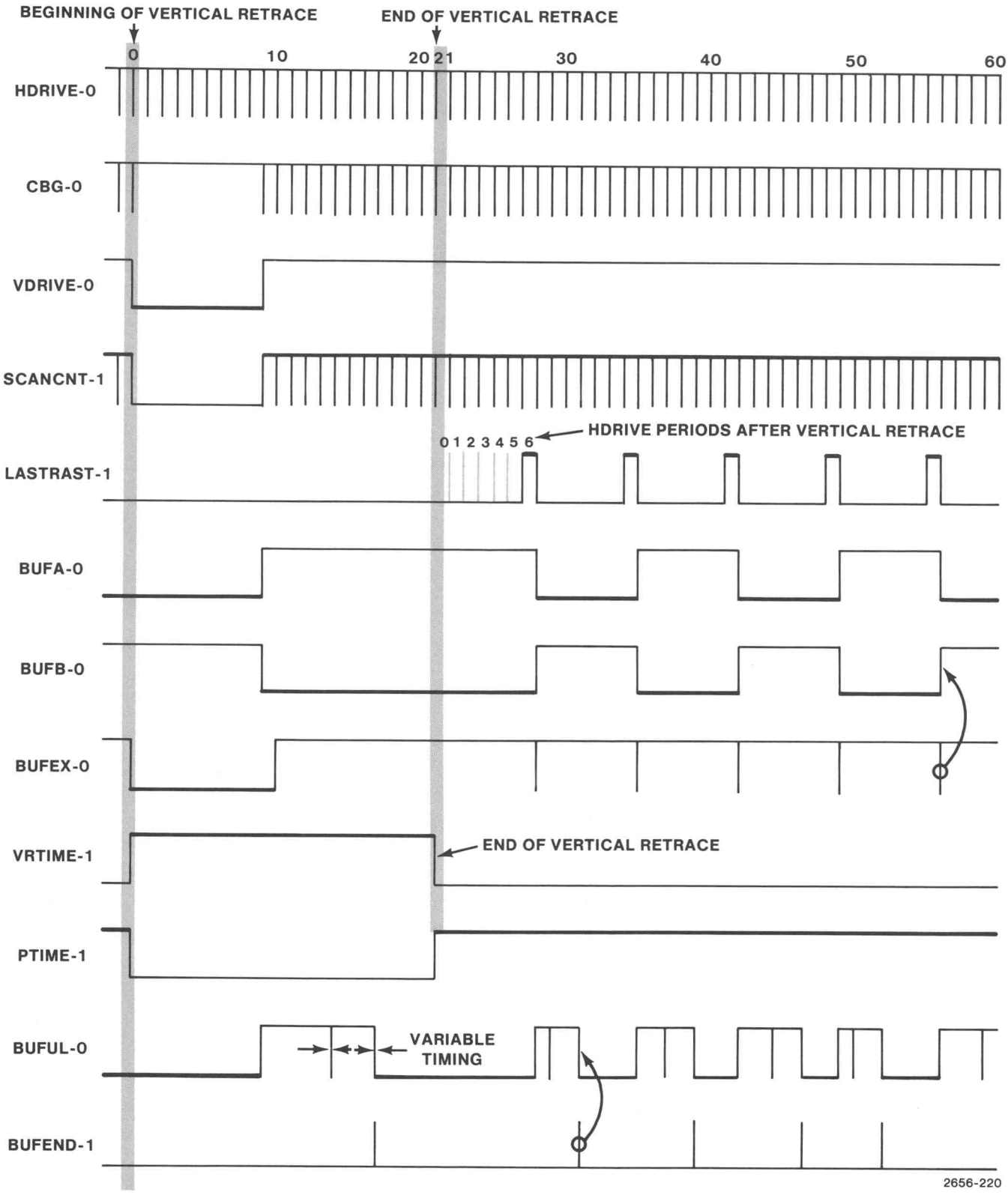


Figure 4-6. Raster Timing, First Six Character Rows.

Raster Line Counter

Refer to Schematic 2-1 and Figures 4-5 and 4-6. The Raster Address Counter Supplies the lower four bits of the addresses that the Display Controller sends to the character memories (RST0-RST3). It also produces LASTRAST-1 (Last Raster), the signal which enables the Alphanumeric Cursor Counter.

The circuit consists of a counter, three gates, and a flip-flop. At the beginning of vertical retrace, the counter is loaded with zeroes. This occurs when PTIME-0 (Picture Time) goes high. As soon as PTIME-0 goes true, at the end of vertical retrace, the counter is allowed to count SCANCNT-0 (Scan Count). Every seventh count thereafter, two gates driven by the counter outputs produce LASTRAST-1. LASTRAST-1, in turn, produces RASTCNTLD-0 (Raster Counter Load) which causes zeroes to be loaded into the counter. At this time, the count cycle starts over.

The three low order bits from the counter, QA, QB, and QC are used for bits 1, 2, and 3 of the raster line address. Bit 0 of the address is produced by a D-type flip-flop (see the lower right corner of Schematic 2-1). This flip-flop changes state only once per video field. The net effect is that the raster address increments by even numbers on one field (0 through 12) and by odd numbers (1 through 13) on the next field. This causes every other byte of a character's pixel data to be retrieved from character memory for successive scan lines, thereby allowing the display to be properly interlaced.

Buffer Control

Refer to Schematics 2-1 and Figures 4-5 and 4-6. The Buffer Control circuit serves four purposes. First, it switches the direction of data flow for the Row Buffers. Second, it signals the Display Memory when a buffer is full. Third, it clears the Buffer Address Counters. Finally, it loads the first byte from the Row Buffers into the Alphanumeric Cursor Counter.

BUFA-0 (Buffer A) and BUFB-0 determine the direction of data flow in Row Buffers A and B, respectively. When BUFA-0 is true, BUFB-0 is false, and vice-versa. If BUFA-0 is true, writing is enabled in Buffer A and it is receiving data from the Display Memory. Simultaneously, Buffer B is being read. BUFA-0 and BUFB-0 are complementary outputs from the same flip-flop. The state of this flip-flop is changed every seventh horizontal scan by BUFEX-0. One Row Buffer supplies its addresses and attribute codes 7 times in succession before it is swapped with the other buffer.

BUFUL-0 (Buffer Full) is bidirectional. Coming from the Display Memory it is a brief, negative-going pulse which clears the font and attribute latches in the Tracker Data Decoder and Latches circuit. The Tracker circuit on the Display Memory Board sends this signal when it detects an end-of-line marker. After an end-of-line marker, the Tracker sends only ASCII spaces, so the font and attribute codes must be reset to zero for the remaining locations in the Row Buffer.

Going from the Display Controller to the Display Memory, BUFUL-0 indicates that the row buffer currently being filled is full. BUFUL-0 is generated by a D-type flip-flop. BUFEND-1 (Buffer End) causes the preset input of this flip-flop to be pulled low, thereby asserting BUFUL-0. BUFEND-1 is generated by the Buffer Address Counter after 81 characters have been written into the buffer. BUFUL-0 goes false at the end of the last raster scan line for a character row. This occurs in coincidence with the positive transition of RASTCNTLD-0 after the Raster Address Counter has been reset to 0. BUFUL-0 also goes false on the first CBG-0 after VDRIVE-0. As soon as BUFUL-0 is false, the Tracker can start transmitting data to the Row Buffers.

LINECLR-0 (Line Clear) is used to clear the Row Buffer Address Counter at the end of each scan line, after the buffer has supplied its addresses for the 80 character columns. LINECLR-0 is asserted on the positive-going edge of LINEND-1. LINEND-1 presets a flip-flop in the Buffer Control circuit which produces LINECLR-0. LINECLR-0 remains true until the first negative transition of COLCLK-1 (Column Clock) after a retrace period.

CURSLD-0 enables loading of the cursor byte from the Row Buffer into the Alphanumeric Cursor Counter. CURSLD-0 is triggered true by LINEND-1 and remains true until the first positive transition of COLSYNC-0 after the retrace period.

BLANK-0 is used to turn off the red, green and blue video data outputs from the Color Map circuit during retrace times. BLANK-1 serves as one of the controlling inputs to the Color Map Address Multiplexer. BLANK-1 and BLANK-0 go true one COLSYNC-0 period after LINEND-1. This allows a two-character delay to provide the time necessary to access the data for the last character in the row and present it before turning off the video. BLANK-0 and BLANK-1 remain true until one COLSYNC-0 period after CURSLD-0. Again, this provides a two-character delay which allows the data for the first character in the row to be accessed before turning on the video.

Buffer Control Multiplexer

Refer to Schematic 2-2 and Figure 5-4. The Buffer Control Multiplexer consists of two quadruple, two-line-to-four-line multiplexers. Its purpose is to switch the buffer control signals BUFCLR-0, LINECLR-0, COLCLK-1, and TCHARCLK-0 (Tracker Character Clock) back and forth between Row Buffers. When BUFA-0 is true, Row Buffer A is receiving data from the Display memory. At the same time BUFA-0 causes the "A" inputs to the multiplexer to be passed on to the outputs. This means that TCHARCLK-0 becomes ACNT-1 (A Count), BUFCLR-0 becomes ACNTCLR-0 (A Counter Clear), and ABUFEND-1 (A Buffer End) becomes BUFEND-1 at the multiplexer outputs.

Similarly, when BUFA-0 is false, Row Buffer A is being read. In this case, COLCLK-1 becomes ACLK-1, LINECLR-0 becomes ANTCLR-0, and ABUFEND-1 becomes LINEND-1 at the multiplexer outputs.

Address Counters A and B

Refer to Schematic 2-2 and Figure 5-4. The purpose of the Address Counters is to increment the Row Buffers through 81 addresses as they either receive or supply data. When the Row Buffer is receiving data, the counter is clocked by TCHARCLK-0 (Tracker Character Clock), which is derived from TCLK-1. When the Row Buffer is being read, the clock is derived from COLCLK-1. COLCLK-1 is synchronous with the display raster; one COLCLK-1 cycle being one character cell wide. The clock signals for Address Counters A and B are ACNT-1 (A Count) and BCNT-1, respectively.

Initially, the counters are loaded with a count of 48. They are then incremented by the appropriate clock through 81 counts. Upon reading the count of 129, an AND gate produces ABUFEND-1 (A Buffer End) or BBUFEND-1. These signals, in turn, produce either BUFEND-1 or LINEND-1 at the outputs of the Buffer Control Multiplexer. For example, if Row Buffer A is receiving data, ABUFEND becomes BUFEND-1 at the output of the multiplexer.

BUFEND-1 or LINEND-1 propagate through the Buffer Control logic and come back to the Buffer Control Multiplexer as BUFCLR-0 or LINECLR-0. They pass through the multiplexer and clear the appropriate counter. Notice that ABUFEND, BBUFEND, LINEND, and BUFEND are true only long enough for the signals to propagate through the logic and clear the counters.

Row Buffers A and B

Refer to Schematic 2-2 and Figure 5-4. The Row Buffers hold character addresses, font addresses, visual attribute codes, and the alphanumeric cursor location. Each Row Buffer consists of a pair of 8 bit by 128 word RAMs. The RAMs are operated in parallel to provide a 16 bit word length. Only 81 of the 128 addresses are used to hold data.

When BUFA-0 is true, writing is enabled in the RAMs for Row Buffer A. They are being written into by the Display Memory as the clock signal. A LOAD-0 switches in step with TCHARCLK-0. ACNT-1 also switches in step with TCHARCLK-0. This causes the address counter to increment after each character has been loaded.

When BUFA is false, data is being read from Row Buffer A. In this case, ALOAD-0 is held low by the Buffer Control Multiplexer and ACNT-1 switches in step with COLCLK-1.

The operation of Row Buffer B is identical to that of Row Buffer A except that when Buffer A is being loaded, Buffer B is being read, and vice-versa.

Tracker Data Decoder and Latches

Refer to Schematic 2-2 and Figure 5-4. Tracker Data comes to the Display Controller as a series of 8-bit bytes. These bytes are presented synchronously with a negative-going clock signal, TCLK-1. Character addresses are interspersed with font addresses and visual attribute codes. Font and visual attribute are sent just ahead of the characters to which they apply. The purpose of the Tracker Data Decoder and Latches circuit is to examine the 8-bit bytes of display list data coming from the Display Memory and to assemble them in 16-bit words in the Row Buffers.

The decoder circuit looks at the upper 4 bits of the display list data (T4-T7) and at TCLK-1. If a font address is present, either the "Y5" or the "Y4" output will be pulsed low. This causes the data on lines T0 through T4 to be loaded into the font latch. This data becomes FONT0 through FONT7 at the latch outputs. If a visual attribute is present at the decoder inputs, the "Y3" output is pulled low. When the Y3 output goes low, the data on T0 through T3 is loaded into the attribute latch and becomes ATT0 through ATT3. When a character address is present, TCHARCLK-0 (Tracker Character Clock) goes true. TCHARCLK-0, after passing through the Buffer Control Multiplexer to become ALOAD-0 or BLOAD-0, causes the byte present on T0 through T6 to be loaded into a Row Buffer. Simultaneously, the content of the font and attribute latches is loaded into the same location in the Row Buffer. When the Tracker circuit encounters an end-of-line marker in the display list, indicating that there are no more characters in the row being transmitted, it pulses the BUFUL-0 line low. The resets the font and attribute latches to font 0 and standard attribute, respectively. The Tracker then transmits ASCII "Space" character addresses until the Row Buffer is full.

A and B Drivers

Refer to Schematic 2-2 and Figure 5-4. The purpose of Drivers A and B is to steer data from the display memory into the row buffers or from the row buffers out to the character memories. Each driver consists of four octal transceivers. When BUFA-0 or BUFB-0 is true, data is allowed to flow from T0 through T7 and from the font and attribute latches into the Row Buffers. When these signals are false, the data in the Row Buffers appears on the lines going out to the Graphics Memory Controller Board.

Shift Registers A, B, and C

Shift Registers A, B, and C receive parallel 8-bit bytes of character cell data from the character memories. They then convert the data to serial bit streams which select color map addresses (and therefore colors) for each pixel in the display.

Each byte of character cell data is loaded into its respective shift register by BITSYNC-0. The shift registers are loaded once for each of the 80 characters in a row. Once the data has been loaded into the shift registers, it is clocked out serially by BITCLK-0.

For graphics characters, all three registers are used. For ROM characters, only Shift Register C contains meaningful data.

Color Map Address Multiplexer

Refer to Schematic 2-2 and Figure 5-4. The Color Map Address Multiplexer steers a 3 bit address from one of four sources to the Color Map. These sources are:

Source 0: This source is selected when displaying color graphics. The address consists of the three serial outputs of Shift Registers A, B, and C. Source 0 is alway selected whenever RAM data is present on the data lines from Graphics RAM (RAMFLAG-0 is true).

Source 1: This source is selected during retrace periods (when BLANK-1 is true). It consists of the outputs of the Color Map Address Latch. During retrace, data in the Color Map Data Latch can be transferred into the Color Map thereby changing the color definitions.

Source 2: This source is always binary 111 and it selects color 7 in the Color Map. Color 7 is the standard background color.

Source 3: This consists of the low order bits (ATT0-ATT3) of the visual attribute codes coming from the Row Buffers. These three bits determine which of the eight colors in the Color Map will be used to display the standard ROM characters. Whenever ROM data is returned from character memory and RAMFLAG-0 is false, the serial output of Shift Register C causes the Color Map Address Multiplexer to switch between Source 2 and Source 3 (i.e., between the background and foreground). A fourth visual attribute bit, (ATT3), when high, causes the sequence of switching to be reversed. This displays the characters with inverted video (the characters are displayed in color 7 on a background determined by the low order bits of the visual attribute code).

Color Map

Refer to Schematic 2-4 and Figure 5-4. The Color Map is essentially a small RAM. It is 6 bits wide by 8 words long. The data which determines the actual color of each pixel is stored here. A 3-bit address to the Color Map RAM selects one of the eight words stored in the RAM. Of the six bits of data stored at each location, two bits control the intensity of the red electron gun, two bits control the green electron gun, and two bits control the blue electron gun. 64 combinations of red, green, and blue are possible. This is the range of colors that the terminal can produce. However, since only eight locations can be selected in the Color Map, only eight colors are displayable at one time. The Processor must change the data in the Color Map when a different color is needed within the eight addressable color categories (this can occur only during retrace periods).

The Color Map receives its data from the Processor by way of the Color Map Data Latch and the I/O port. The outputs of the latch pass over data lines CMD0 through CMD5.

The output of the color Map RAM goes to a tri-state latch. This latch is clocked by BITCLK-1 so that the color map data transitions are synchronized to BITCLK-1.

Blanking is accomplished by passing the red, blue, and green data through six NAND gates which are disabled when DBLANK-0 is true. DBLANK-0 is derived from BLANK-0 by passing BLANK-0 through an inverter and then through a latch which is clocked by BITCLK-1. This synchronizes the transitions of DBLANK-0 to BITCLK-0. Notice that in the blanking state, the data outputs all go high.

Delay Latches

See Schematic 2-4 and Figure 5-4. The Delay Latches delay the visual attribute codes (ATT0-ATT3) and the Cursor Signal (CUR-1) for two character cell periods, and RAMFLAG-0 for one character cell period. This is necessary because it takes this amount of time for character data to be accessed in the character memories and loaded into the Shift Registers. RAMFLAG-0 need only be delayed one character cell period because it is sent from the Graphics Memory Controller with the character data. The delay is accomplished by loading the cursor and attribute data through two serially-connected 4 bit latches on two successive COLCLK-1 cycles. RAMFLAG-0 passes through only one of the latches.

Video Output

Refer to Schematic 2-4 and Figure 5-4. The Video Output circuit supplies five video signals for use by external video equipment. These are red, blue and green Composite Video, Monochrome Composite Video, and Hard Copy Video (see Table 4-2).

Table 4-2

HARD COPY AND VIDEO OUTPUTS

Labeled	Jack No.	Cable Color	Description
MONOCHROME	J5300	white	monochrome video
RED	J5600	red/white	red composite video
GREEN	J5700	green/white	green composite video
BLUE	J5800	blue/white	blue composite video
SYNC	J5500	orange/white	composite sync
BLANK	J5400	brown/white	composite blanking

There are three digital-to-analog converters (DACs): one for each color video output. Each DAC receives two bits of color data from the Color Map. It processes this data using two inverters, a NAND gate, and a resistive summing network to produce four discrete voltage levels. These voltage levels appear at the input to an emitter follower. Two additional inputs, SYNC-1 and CURSOR-0, add composite sync and cursor information to the video. The output of the emitter follower does two things: first, it drives a video output stage which shifts the level of the video and provides a 75-ohm output impedance. Secondly, it drives two mixing circuits which combine red, green, and blue video to produce the monochrome and hard copy outputs.

Alphanumeric Cursor Counter

See Schematic 2-3 and Figure 5-4. The first byte out of the Row Buffers contains the location of the alphanumeric cursor. This byte is loaded into the Alphaumeric Cursor Counter by CURSLD-0. LASTRAST-1 enables the counter during the last horizontal scan line for a character row. Once enabled, the counter decrements each time COLCLK-1 goes true. When the counter reaches zero, ACUR-0 goes true. ACUR-0 goes to the cursor control circuit and, after processing there, causes the display to be intensified for one character cell width.

In order for the cursor to appear in a character row, the cursor counter must be loaded with a value less than 80. If the value is greater than this, the counter cannot reach zero before the end of the scan line. A cursor value less than 80 is stored in the display list for one row only, and this is always a row which is in view on the screen.

X and Y Cursor Counters

Refer to Schematic 2-3 and Figure 5-4. The X and Y Cursor Counters are used to control the position of the graphics cursor.

The Y Cursor Counter is loaded with position data from the Y Cursor Latch. This occurs during vertical retrace, on the negative transition of PTIME-1. After vertical retrace, the counter is decremented by SCANCNT-1 on each horizontal scan. When the counter reaches zero, YCUR-0 goes true. YCUR-0 is then used by the Cursor Control logic to intensify one complete horizontal scan line.

There are 476 possible locations for the Y cursor (one for each scan line in the 34 character rows). If the Y cursor latches are loaded with a value greater than 476, the cursor does not appear on the display because the counter cannot reach zero. This is the method by which the processor controls whether the Y cursor appears on the display.

An additional input to the Y cursor counters consists of the EXCLUSIVE OR functions of the low-order bit from the Raster Address and the low-order bit of the Y cursor position data. This input enables the counter to count only on the even or the odd raster field, depending on whether the cursor is to appear on an even or an odd line. For example, if the cursor is to appear on line 379, the counter only counts down when an odd line is acanned (RAST01). This allows two modulo-16 counters (with a maximum count state of 256) to account for 480 cursor positions.

The X cursor counter consists of three modulo-16 counters. It is loaded with position data from the X Cursor Latches during each horizontal retrace (by CBG-0). The X cursor can have one of 640 locations. If the processor loads a value greater than this into the X cursor latches, the X cursor does not appear.

The X cursor is decremented by BITCLK-0. When it reaches zero, XCUR-1 causes the display to be intensified for one BITCLK-1 period (one pixel) at that point.

Processor Input/Output

Refer to Schematic 2-3 and Figure 5-4. The processor communicates with the Display Controller by way of the I/O controL, I/O Bus Driver, and Display Controller Status Driver. All data coming into the Display Controller is loaded into one of six latches. These are the Color Map Address Latch, the Color Map Data Latch, the two X Cursor Latches and the two Y Cursor Latches. In addition, the Processor can read certain status information through the Display Controller Status Driver. The Display Controller has six addresses on the Processor Bus. Table 4-3 lists these along with their functions.

The I/O control circuit examines bus address lines BA0 through BA7. It also receives IOADR-1. Whenever a Display Controller address is presented and WRITE-0 is true, one of the following signals is asserted: YCURLO-0, YCURHI-0, XCURLO-0, XCURHI-0, MAPADR-0, and MAPDAT-0. The positive-going edge of these signals causes the data present on the data bus to be loaded into their respective registers. The Color Map Address Latch and the Display Controller Status Driver have the same bus address. However, MAPADR-0 becomes active only when WRITE-0 is true and STATUSEN-0 becomes true only when READ-0 is true.

When data is written into Color Map Data Latch by the processor, the least significant bit of the Status Word is set by MAPDAT-0. The processor reads this bit to determine when to send new data. The first CBG-1 that comes along after the data has been sent causes the data to be loaded into the Color Map RAM by enabling the NAND gate that produces CMCD-0. Simultaneously this clears bit 0 in the Status Driver.

VRTIME-1 can be read through bit 1 of the Status Driver. The Processor uses this bit to determine when vertical retrace is occurring.

Table 4-3

DISPLAY CONTROLLER I/O ADDRESS MAP

Name	Address	Bit	Read Function	Write Function
Control and Status	X'8E0'	0	1=Color Map busy	Color Map address bit 0
		1	1=vertical retrace	Color Map address bit 1
		2	none	Color Map address bit 2
		3-7	none	none
Color Map Data	X'8E1'	0-1	none	blue data
		2-3	none	green data
		4-5	none	red data
X Cursor Low	X'8E2'	0-7	none	low-order bits of cursor location
X Cursor High	X'8E3'	0	none	MSB of cursor location
		1-7	none	none
Y Cursor Low	X'8E4'	0-7	none	low order bits of cursor location
Y Cursor High	X'835'	0	none	MSB of cursor location
		1-7	none	none

Cursor Control

Refer to Schematics 2-3 and 2-4 and Figure 5-4. The Cursor Control receives the three cursor counter outputs ACUR-0, XCUR-0, and YCUR-0 and processes them to generate CURSOR-1 and CURSOR-0.

ACUR-0 and YCUR-0 go to the same OR gate. The output of this gate is CUR-1. CUR-1 passes through the Delay Latches where it is delayed for two COLCLK-1 periods. After the delay, CUR-1 is EXCLUSIVE ORed with XCUR-1 passed on to a latch. The latch synchronizes the cursor signal transitions to BITCLK-1.

CURSOR-0 goes to the DACs in the Video Output circuits and Z-Axis circuits. CURSOR-1 disables the outputs of the Color Map during the cursor period.

Hard Copy Control

Refer to Schematic 2-3 and Figure 5-4. The purpose of the Hard Copy Control is to provide "handshaking" between the terminal and the Video Hard Copy Unit. It consists of a one-shot, three inverters, and a driver transistor.

At the beginning of a hard copy operation, the Hard Copy Control receives a pulse on the HC-0/BUSY-0 line from the Display Memory Board. This triggers the one-shot. The one-shot turns on the driver transistor (through an inverter) causing the COPY-0 line to go true, thereby triggering the Hard Copy Unit. At the same time, the one shot (through a second inverter) pulls down the HC-0/BUSY-0 line. The period of the one-shot is set long enough to hold down the HC-0/BUSY-0 line until the Hard Copy Unit responds with BUSY-0. The Hard Copy Unit then holds the line low until it is finished making the hard copy.

GRAPHICS MEMORY CONTROLLER BOARD

The block diagram for the Graphics Memory Controller Board is Figure 5-5.

The Graphics Memory Controller performs three primary functions. Its first and most important function is to fetch RAM and ROM character data for the Display Controller, synchronously, as the display screen is scanned. Secondly, it enables the Processor to read RAM and ROM or to write RAM character data as it is defining graphic cells. Thirdly, it "refreshes" the Graphics RAM.

Figure 4-7 depicts the character memory address space. The RAM array is 24 bits wide and may contain up to 64K words (32, 2K "fonts"). Each of the words is divided into three 8 bit sections called "Planes" A, B, and C.

If the Character Set Expansion Board is installed, the ROM array may be as large as 16K words by eight bits (eight "fonts" of 2K each). When a ROM character is read by the Processor or the Display Controller, eight bits of data are transferred over the same paths as Plane C RAM data.

Enough RAM may be installed to overlap the ROM addresses. In this case, ROM takes precedence over RAM. In order to conserve memory, RAM fonts start at the top, while ROM fonts start at the bottom of character memory space.

Simultaneous operations in both arrays are allowed. For example, if the Graphics Memory Controller is busy reading a RAM character for the Display Controller, the Processor may access ROM (or vice versa). The Display Controller may directly address the memory arrays any time during a horizontal scan line. During horizontal retrace, or vertical retrace, access by the Display Controller is inhibited. Since the Processor is not synchronized to the raster display, it may request access to the RAMs at any time. However, the access may be delayed if the memory is busy with the Display Controller.

The Processor accesses character data through several I/O ports. To do this, it must write an address consisting of font (five bits), character (seven bits) and raster (four bits) into three of the I/O ports. Then, depending on whether a read or a write is being performed, the Processor must read or write three 8-bit bytes of data at additional I/O register addresses on the Graphics Memory Controller Board.

The Processor is synchronized to the Graphics Memory Controller by means of a ready flag in its Control and Status Word. When the ready flag is true, the system has no pending I/O operations and the Processor may change addresses and/or write data. Once the Processor has set up the desired addresses and data, the operation is initiated by setting a read or write trigger bit in the Control and Status Word. When this is done, the ready flag goes false. The ready flag remains false until the I/O operation is complete.

The amount of time it takes for the Processor to access a particular row within a character varies depending on how busy the Display Controller is. If a small number of characters is being displayed, the ready flag may become true in the minimum time (1.5 μs). If the display is busy, it may take as long as one complete scan line (63.4 μs).

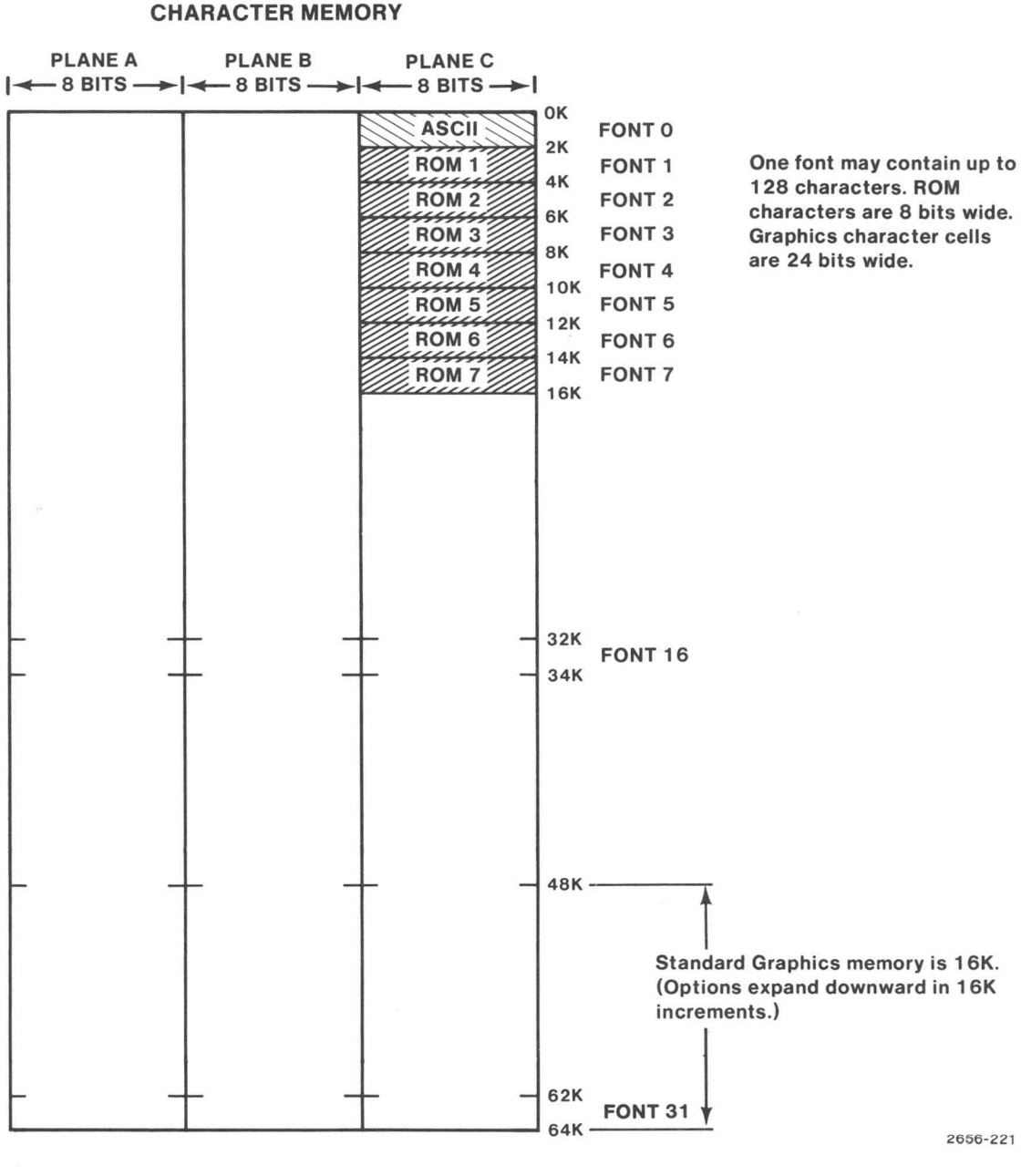


Figure 4-7. Character Memory Address Space.

I/O Port Address Decoding and Data Bus Transceiver

Refer to Schematic 3-1 and Figure 5-5. The I/O Port Address Decoding Logic looks at the Processor address bus and determines whether or not an I/O operation is directed to the character memories. An eight input NAND gate and three associated inverters decode the most significant bits of the address (including IOADR-1) and generate a logic low when a character memory address has been asserted. This enables a 3-line to 8-line decoder which examines the lower four bits of the address. This decoder activates one of seven outputs which correspond to individual word addresses within the Graphics Memory Controller. These addresses are the I/O Port Memory Address Latches, the I/O Port Read Data Latches, the I/O Port Write Data Latches and the I/O control and status word. See Table 4-4 .

The outputs from the 3-line to 8-line decoder, along with READ-0 or WRITE-0 are processed by several gates to generate the following control signals:

MYREAD-0 controls the direction of the Data Bus Transceiver. If the character memory is being addressed and READ-0 goes low, MYREAD-0 also goes low. When this occurs, character memory data is placed on the data bus to the Processor.

CONTROLRD-0 (Control Read) enables one section of a data bus driver in the Control and Status Word, thereby asserting the status of IORQ-1/RDY-0 on the data bus.

CONTROLWR-0 (Control Write) enables the Control and Status Word circuit, thereby enabling I/O port read/write operations.

FONTRD-0 (Font Read) enables one section of a data bus driver in the Control and Status Word. This allows the Processor to read the outputs of the I/O Port Font Decoder to see how much RAM and ROM are installed.

FONTLD-0 (Font Load), CHARLD-0 (Character Load), and RASTLD-0 (Raster Load), which are write strobes. They latch font, character and raster addresses into the I/O Port Memory Address Latches.

PARD, PBRD, and PCRD-0 (Plane A Read, etc.), which cause the Plane A, B and C Data to be latched into the I/O Port Read Data Latches.

PAWR, PBWR, and PCWR-0 (Plane A Write, etc.), which cause data from the I/O data bus to be latched into the I/O Port Write Data Latches.

Table 4-4

GRAPHICS MEMORY CONTROLLER I/O ADDRESS MAP

Name	Address	Bit	Read Function	Write Function
Font Address	X'870'	0	1=RAM Font installed	low-order bits of font adress
		1	1=ROM Font installed	bit 1 of the font address
		2-4	none	font address MSBs
		5-7	none	none
Character Address	X'871'	0-6	none	character address
		7	none	none
Raster Address	X'872'	0-3	none	raster line address
		4-7	none	none
Plane A data	X'873'	0-7	Plane A character data	Plane A character data
Plane B data	X'874'	0-7	Plane B character data	Plane B character data
Plane C and ROM data	X'875'	0-7	Plane C and ROM character data	Plane C and ROM character data
Control and Status	X'876'	0	I/O ready	none
		1-2	none	none
		3	none	write trigger
		4-6	none	none
		7	read trigger	none

I/O Port Memory Address Latches and Drivers

Refer to Schematics 3-1 and 3-2 and Figure 5-5. The I/O Port Memory Address Latches derive their inputs from the data bus lines D0 through D7. Their outputs form a 16-bit address by which the Processor accesses the character memories. One latch stores the font address, one the character address, and one the raster address. The addresses are clocked into their respective latches by FONTLD-0, CHARLD-0, and RASTLD-0.

The outputs of the latches may be directed onto either the RAM or the ROM address bus. The routing is controlled by two tristate drivers; the RAM and ROM Address Drivers. The RAM and ROM Address Drivers are enabled by IORAMS-0 (I/O RAM Select) and IOROMS-0 (I/O ROM Select) which are produced by the Cycle Type Control Logic.

I/O Port Write Data Latches

Refer to Schematic 3-1 and Figure 5-5. The I/O Port Write Data Latches correspond to Graphic Memory planes A, B, and C. These latches are clocked by PAWR-0, PBWR-0, and PCWR-0 (Plane A Write, etc.). They derive their inputs from data bus lines D0 through D7 and their outputs go directly to the RAM array.

RAM/ROM Data Multiplexer

Refer to Schematic 3-2 and Figure 5-5. Data to the I/O Read Data Latch for Plane C is routed through a pair of multiplexers. When reading RAMs, CR0 through CR7 are connected into the Plane C latch. When reading ROMS, R0 through R7 are connected into the latch. The state of the multiplexers is selected by IOROMS-0. This signal selects the ROM data inputs, when true.

I/O Port Read Data Latches

Refer to Schematic 3-2 and Figure 5-5. There are three read data latches for the I/O port; one each for planes A, B and C. The tri-state outputs for the A, B, and C Latches are enabled by PARD-0 (Plane A Read), PBRD-0, and PCRD-0, respectively. Data is clocked into the Latches by IORDCLK-1 (I/O Read Clock). IORDCLK is generated by the Cycle Type Control Logic when an I/O Port read operation is performed. The A and B Latches are straight forward: AR7 data goes into the A Latch and BR7 goes to the B Latch. The Plane C latch performs a double function: it can hold plane C RAM data or it can hold ROM data.

I/O Port Font Decoder

Refer to Schematic 3-2 and Figure 5-5. The I/O Port Font Decoder examines the I/O address bus to determine whether a ROM font or a RAM font is being addressed. It provides output signals which the Processor can read to determine how much ROM or RAM has been installed and enabled in the character memories. It also provides enabling signals which are used in accessing these memories.

A11 through A15 provide input to the Font Decoder. One section of the decoder examines A14 and A15 and enables one of four output lines. These outputs correspond to 16K banks of Graphics RAM. A second section of the decoder examines A11 through A15 and selects one of the eight outputs corresponding to ROM fonts 0 through 7. These outputs go to several pairs of jumper pins which are used to enable or disable the selection of RAM and ROM fonts.

Those outputs of the RAM and ROM font decoders for which jumpers have been installed form a "wire AND" circuit. These wire ANDs control two NOR gates whose outputs produce ROMIN-1 (ROM installed) and RAMIN-1 (RAM installed). Notice that whenever ROMIN-1 is true, RAMIN-1 is disabled. This allows ROM to take precedence over RAM if they overlap into the same address space and are both enabled.

I/O Port Cycle Sync Logic

Refer to Schematic 3-2 and Figure 4-8. In contrast to the Display Controller, which sends addresses synchronously with its own clock, I/O port memory requests need synchronization. This is accomplished in the following way: A flip-flop in the I/O port cycle sync logic receives an IORQ signal from the Control and Status word. It clocks it into its outputs on the following CMPCLK-0 low-going transition. This produces an IORQSTS-1 (I/O Request Synchronized). IORQSTS-1 enables two AND gates. Those AND gates then pass ROMIN-1 or RAMIN-1 to two more flip flops. These are the I/O RAM Request Flip Flop and I/O ROM Request Flip Flop. RAMIN or ROMIN are clocked into these flip flops on the next CMPCLK-0 low transition to produce IORAMQ-1 or IOROMQ-1. The I/O request signals are cleared by IOCDONE-1 after the memory cycle is complete.

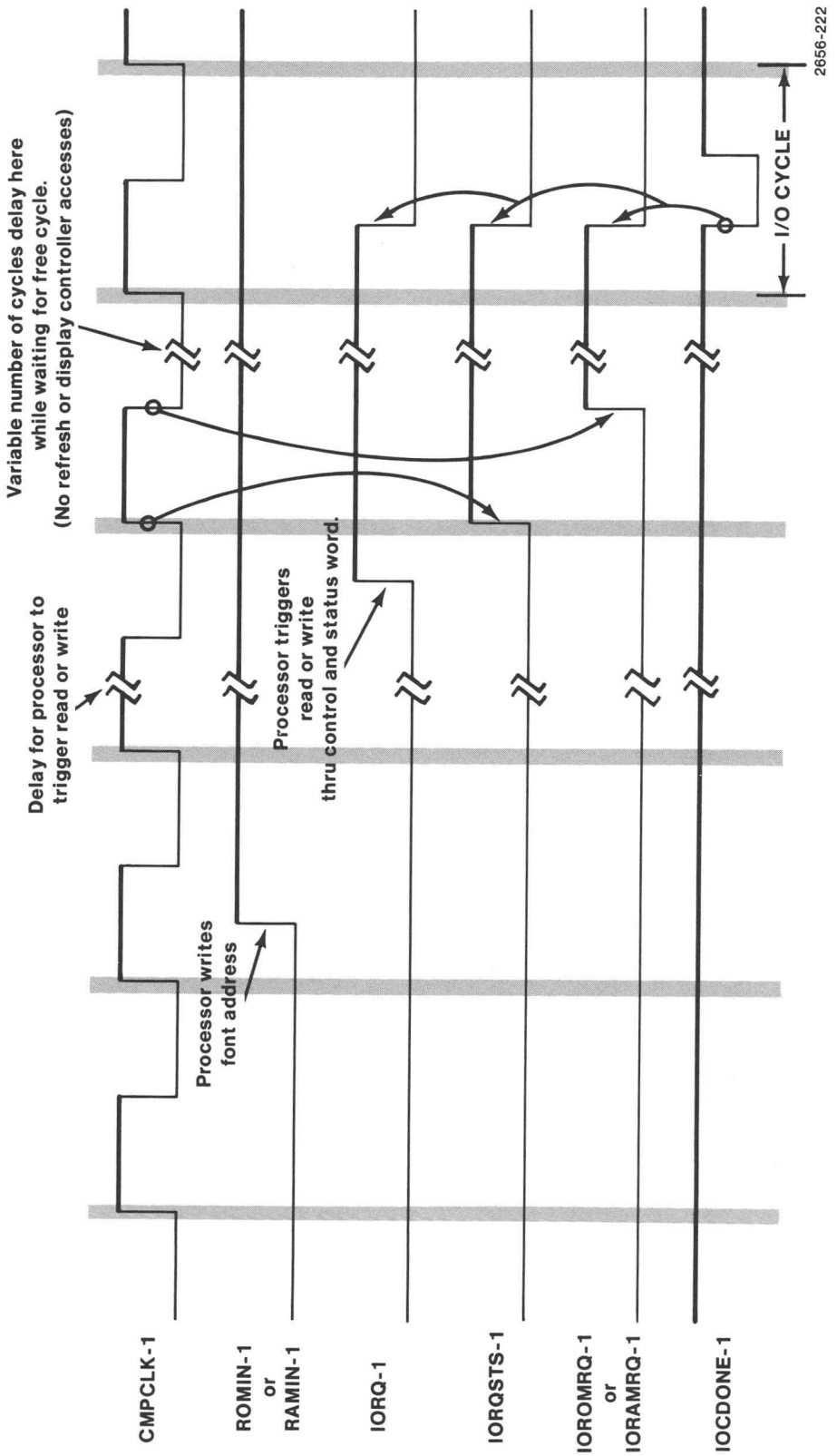


Figure 4-8. I/O Cycle Syncronization.

I/O Control and Status Word

Refer to Schematic 3-1 and Figure 5-5. The I/O Control and Status Word is used by the Processor to initiate read/write operations, to monitor the status of these operations, and to determine how much RAM or ROM is enabled within the character memories.

Data bits 3 and 7 on the I/O Data bus feed into four gates which perform an EXCLUSIVE OR function. Two outputs from the EXCLUSIVE OR circuit, along with CONTROLWR-0 enable or disable a pair of 3-input gates. The outputs of these gates are DOREAD-1 (Do a Read) and DOWRITE-1. These signals are active under the following conditions: When D7 is true and a CONTROLWR-0 occurs, the DOREAD-1 signal is asserted. If D3 is true when CONTROLWR-0 occurs, DOWRITE-1 is asserted. If both D3 and D7 are at the same state, nothing is generated.

An OR gate, whose output is controlled by RAMIN-1 or ROMIN-1 serves to inhibit I/O operations if no memory is installed at the font being addressed.

I/O Interface Flags R-1/W-O (Read/Write), W-1/R-0, RDY-1/IORQ-0, and IORQ-1/RDY-0 (I/O Request/Ready) are produced by two flip-flops formed from 3-input NOR gates. Any time a DOREAD-1 or a DOWRITE-1 occurs, the I/O Request Flip-Flop is set. This asserts IORQ-1 and IORQ-0 (I/O Request). The I/O Request Flip-Flop is reset by IOCDONE-1 (I/O Cycle Done) from the cycle type control logic.

Whenever a DOREAD-1 occurs, the Read/Write Flip-Flop is set. This asserts R-1 and R-0, signalling a read operation. If a DOWRITE-1 occurs, the read/write flip flop is reset, thus asserting W-1 and W-0 and signaling a write operation.

FONTRD-0 (Font Read) enables one half of a data bus driver. This driver places two flag signals on the I/O data bus which the Processor can read. These two flags are derived from ROMIN-1 and RAMIN-1 from the I/O Font Decoder. The Processor then knows whether the font last written into the font address register is RAM, ROM, or not enabled.

CONTROLRD-0 enables the second half of the status driver in the control and status word. This asserts the status of RDY-1/IORQ-0 on to the I/O data bus so that the processor can determine the status of the last I/O operation.

Display Controller Address Latches, RAM Address Drivers, and ROM Address Drivers

Refer to Schematic 3-1 and Figure 5-5. Two latches are loaded with addresses, transmitted from the Display Controller. One pair of 8 bit dat bus drivers allows the outputs of these latches (DA0 — DA15) to drive the RAM address bus (RAM0 — RAM15) while a second pair of drivers allows the latch outputs to drive the ROM address bus (ROM0 — ROM13).

Display Controller Font Decoder and Display Controller Blanking Flip-Flop

Refer to Schematic 3-3 and Figure 4-9. The Display Controller Font Decoder examines the Display Controller address bus lines FONT0 — FONT5 and provides select signals used in accessing the character memories. The operation of the Display Controller Font Decoder is similar to the I/O Port Font Decoder with the following exception: Its outputs may be disabled from one of two sources. First, it can be disabled by the Display Controller Blanking Flip-Flop. The output of the Display Controller Flip-Flop is active during retrace periods when the Display Controller's clock, COLCLK-1 is turned off. The flip-flop is reset by the first HDRIVE-0 after retrace. Secondly, the Font Decoder may be disabled by INHIBIT-1 from the Space Character Inhibit Logic (See below).

The outputs of the Display Controller Font Decoder are DCROMRQ-1 and DCRAMRQ-1 which signal that the Display Controller is requesting access to ROM and RAM, respectively.

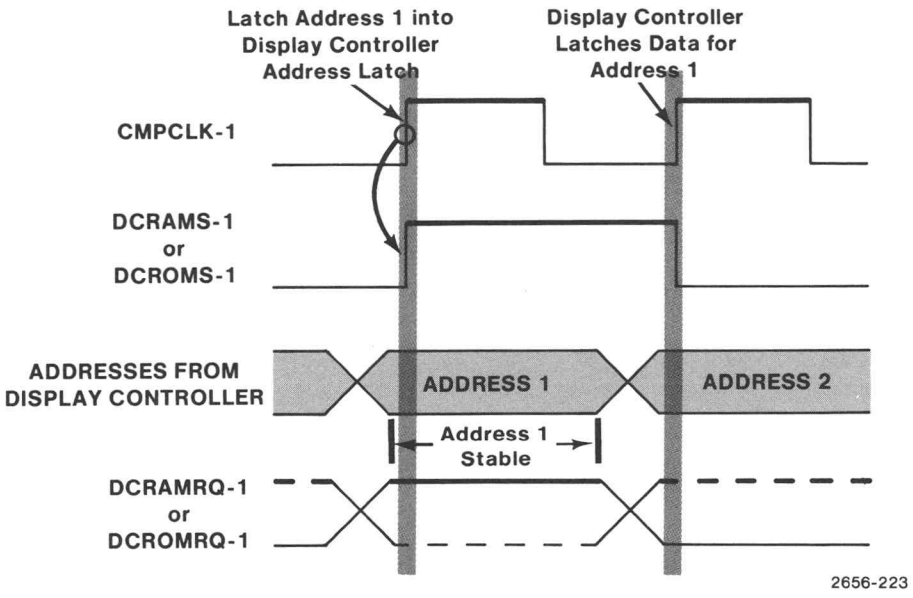


Figure 4-9. Display Controller Memory Access Cycle.

ASCII Space Inhibit Logic

Refer to Schematic 3-3 and Figure 5-5. The Space Inhibit Logic allows the character memory cycles that would normally be used in retrieving ASCII Space characters for the Display Controller to be used for Processor I/O.

Several gates and an inverter decode the font and character address lines from the Display Controller. When font 0 and character address x'40' (Space) are present, INHIBIT-1 goes true. This blanks the Display Controller Font Decoder for the current memory cycle.

Data for the ASCII Space consists of all zeroes. Since the RAM and ROM tristate drivers are both turned off when the D.C. Font Decoder is disabled, their outputs are held positive by the pull-up resistors on these lines. This corresponds to an all zero output to the Display Controller.

Display Controller RAM Data Latches

Refer to Schematic 3-2 and Figure 5-5. When RAM data is being read for the Display Controller, the data is held in three latches. The latch for Plane C has tri-state outputs since it has to be turned off when ROM data is being sent to the Display Controller. The latches for Planes A and B drive the Display Controller at all times. DC RAMS-0 enables the Plane C tri-state outputs.

Display Controller ROM Data Driver

Whenever a ROM character is being read for the Display Controller, the data passes through the Display Controller ROM Data Driver and goes out over the Plane C data paths to the Display Controller (notice that the ROM data is low when true). DCROMS-0 (Display Controller ROM Select) enables the tri-state outputs of the driver.

ASCII ROM

Refer to Schematic 3-4. The ASCII ROM occupies font 0. It contains pixel data for the standard alphanumeric characters and certain special control characters. CS0, from the ROM Font Selector passes through a cut-strap to become ASCEN-0. ASCEN-0 turns on the ASCII ROM Tri-state Driver when font 0 is selected. Bus lines ROM0 through ROM10 address the ROM and the Adder circuit. The Adder receives ROM0 through ROM3. These lines carry the raster line address. The Adder offsets the raster line address by two, thereby dropping the character's data two scan lines within its cell.

Clock Generation Logic

Refer to Schematic 3-3 and Figure 4-10. The purpose of the clock generation logic is to provide a substitute clock signal for I/O port and refresh memory operations during horizontal and vertical retrace. The Display Controller's clock, COLCLK-1, stops during these periods.

The Clock Generation Logic receives the following inputs from the Display Controller: COLCLK-1, the 650 ns memory cycle clock; CBG-0, which occurs near the end of each horizontal retrace period; VDRIVE-0, which occurs at the beginning of each vertical retrace; and HCLK-1, an 18 MHz clock generated on the Processor board.

FCLK-1 is a buffered version of HCLK-1. The Substitute Clock Counter divides FCLK-1 by 16 to produce an 870 ns clock signal, SUBCLK-1 (Substitute Clock). During horizontal scan, the Substitute Clock Counter is continuously cleared by SUBCLKEN-0 (Substitute Clock Enable) before it can count enough FCLK-1 cycles to produce an output (SUBCLKEN-0 follows CHARCLK-0 which is an inverted version of COLCLK-1).

When horizontal retrace begins, COLCLK-1 stops at the logic high state. This causes SUBCLKEN-0 to be held low. The Substitute Clock Counter now counts FCLK-1 and produces SUBCLK-1 (Substitute Clock). SUBCLK-1 is inverted to produce SUBCLK-0. This signal appears at one input to a complementary output NAND gate (U225C). The other input to the NAND gate is CHARCLK-1 (a buffered COLCLK-1). The NAND gate "combines" the two signals to produce CMPCLK-1 and CMPCLK-0 (Composite Clock). These signals have periods of 650 ns during scan periods and 870 ns during retrace.

Before the end of the retrace period, it is necessary to shut down the Substitute Clock Counter. This is done in the following way: When CBG-0 goes low, the J input of the Shut Down Flip-Flop is forced high and K input is forced low. As a consequence, on the next low-going transition of SUBCLK-1, STOPCLK-1 goes high. This causes SUBCLKEN-0 to be held high, thereby disabling the Substitute Clock Counter. The Shut-Down Flip-Flop is cleared by CHARCLK-0 when the Display Controller's clock starts up again. At this time, SUBCLKEN-0 begins following CHARCLK-1 and clearing the Substitute Clock Counter.

During vertical retrace, SUBCLK-1 shutdown is held off until the 13th CBG-0 following VDRIVE-0. At the beginning of vertical retrace, VDRIVE-0 loads a 4 bit counter with a value of 2. After the counter counts 13 CBG-0 cycles, its Max/Min output goes high. This enables NAND gate U225B, allowing CBG-0 to pass on to the Shut Down Flip-Flop and initiate the shut down sequence.

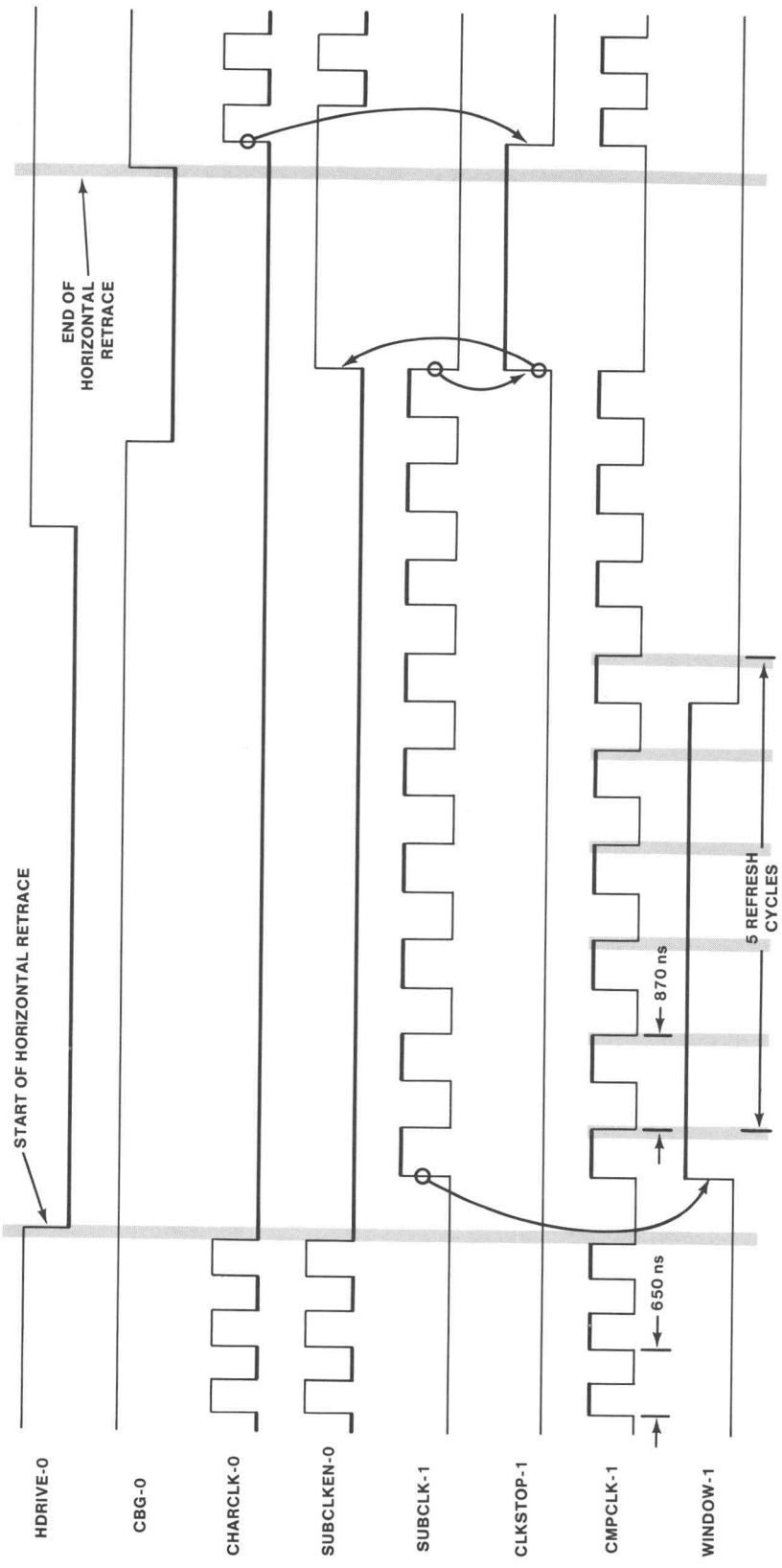


Figure 4-10. Graphics Memory Horizontal Retrace Timing.

Cycle Type Control Logic

Refer to Schematic 3-3. At the input to the Cycle Type Control Logic there are six NAND gates which function as cycle priority gates. These gates arbitrate priorities among the various kinds of memory cycle requests. For example, a Display Controller RAM Cycle request causes the cycle priority gates to disable I/O Port RAM requests. Four signals provide inputs to the cycle priority gates: DCRAMRQ-1; DCROMRQ-1; IORAMRQ-1; and IOROMRQ-1.

The outputs of the gates are clocked into a latch on each CMPCLK-1 positive transition. This generates a set of state signals which identify the memory cycle type. These state signals are IOROMS (I/O ROM Select), IORAMS (I/O RAM Select), DCROMS (Display Controller ROM Select), and DCRAMS (Display Controller RAM Select). The state signals are routed to various parts of the Graphics Memory Controller for controlling data flow.

IORDCLK-1 is produced as the AND condition of R-1 (Read) from the control and status word, DATACLK-1 (Data Clock) from the RAM Sequencer, and an I/O state signal from within the cycle-type logic. It causes data in the I/O Port Read Latches to appear on the data bus.

RAMFL-0 (RAM Flag) is sent to the Display Controller whenever a RAM operation is taking place. This allows the Display Controller to know that it is receiving graphics data from RAM.

IOCDONE-1 (I/O Cycle Done) is produced from CYCLK-1 (Cycle Clock) from the RAM Sequencer (CYCLK-1 occurs near the end of any RAM cycle) in combination with two cycle type signals; IOROMS-0 and IORAMS-0.

Two additional state signals REFRESH-1 and REFRESH-0 are generated by the Refresh Flip-Flop. The state of WINDOW-1 is clocked into the flip-flop by CMPCLK-0. WINDOW-1 is true for five CMPCLK-0 cycles, at the beginning of horizontal retrace periods. REFRESH-0 enables the refresh function of the Refresh Counter.

Refresh Window Logic

Refer to Schematic 3-3 and Figures 4-10 and 4-11. Since the Graphics Memory consists of dynamic RAMS, they must be refreshed periodically. The Refresh Window Logic generates a time window near the beginning of the horizontal retrace period for this purpose. A modulo-16 counter is loaded with a preset count of 9. When HDRIVE-0 goes true, the counter counts SUBCLK-1 until both max/min and HDRIVE-0 go high (this occurs after 5 positive transitions of SUBCLK-1). At this time, the window flip-flop (U235B) is cleared and the counter is disabled (its G input is forced high). This sequence generates WINDOW-1 and WINDOW-0. WINDOW-1 and WINDOW-0 are five SUBCLK cycles long and they cause five memory refresh cycles to occur.

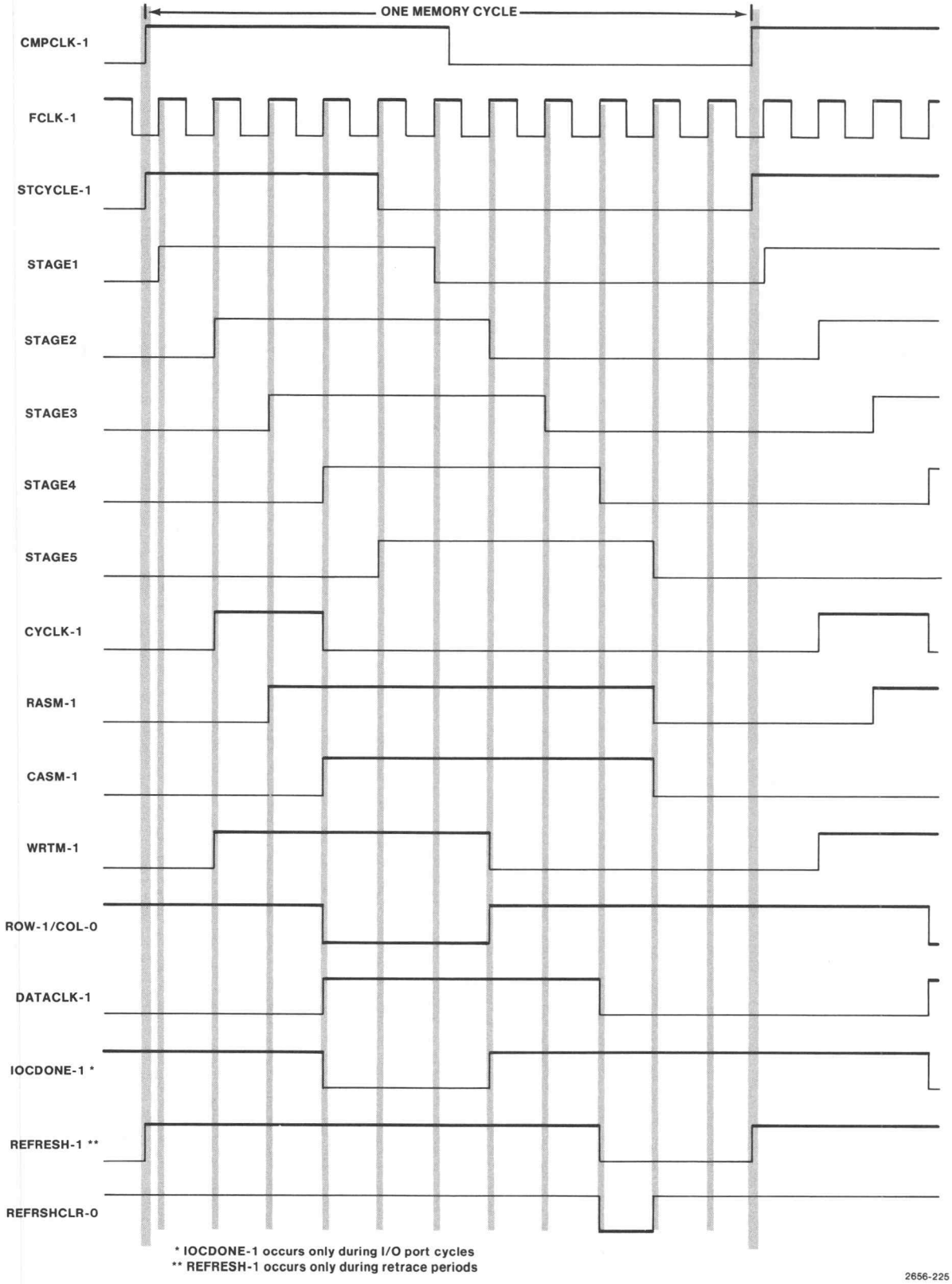


Figure 4-11. RAM Sequencer Waveforms.

RAM Sequencer

Refer to Schematic 3-3. The timing portion of the RAM sequencer is made up of two D-type flip-flops and four sections of a latch. Together, they constitute a five-stage shift register. An additional flip-flop serves as an input stage to the Shift register.

Figure 4-11 shows the RAM Sequencer Waveforms. A RAM cycle occurs once every CMPCLK-1 period. At the beginning of a memory cycle, the Q-0 outputs of the shift register are all cleared. A logic 1 is loaded into the Q-0 output of the input stage by CMPCLK-1. This 1 then propagates through the five stages of the shift register in step with FCLK-1. The Q-0 output of the last stage is connected back to the preset terminal of the input stage. This causes a 0 to be shifted back through the shift register on the next five FCLK-1 cycles. The sequence of shift register states is decoded by several gates to produce a series of signals which control the Graphics Memory RAMs during memory cycles. These signals are:

- RASM-1 (Master Row Address Strobe). This signal synchronizes the loading of the row address within the RAM devices.
- CASM-1 (Master Column Address Strobe). This signal synchronizes loading of the column address within the RAM devices.
- ROW-1/COL-0 (Row/Column Select). This signal determines whether the address being presented to the RAM devices is a row or a column address.
- DATACLK-1 (Data Clock). This signal latches RAM output data into the Display Controller RAM data latches. It is also used in the Cycle Type Control Logic to enable IORDCLK-1, which latches data into the I/O port read data latches.
- CYCLK-1 (Cycle Clock). This timing signal occurs near the beginning of a Memory cycle and is used in the Cycle Type Control Logic to generate IOCDONE-1.
- RFRSHCLR-0 (Refresh Clear) clears the Refresh Flip-Flop.
- WRTM-1 (Write Master). When true, WRTM-1 enables writing into the RAM's. When it is not true, only read operations take place.

ROM Font Selector

Refer to Schematic 3-4 and Figure 5-5. The ROM font selector looks at the top 3 bits of the ROM address bus. According to the state of those bits, it selects one of eight chip select signals, CS0-0 through CS7-0. These signals correspond to ROM fonts 0 through 7. CS0-0 goes to the ASCII ROM as ASCEN-0. All eight go to the Character Set Expansion board. The selector outputs are enabled by IOROMS-0 or DCROMS-0 whenever a ROM cycle is occurring.

RAM Address Multiplexer and Refresh Counter

Refer to Schematic 3-4 and Figure 5-5. The RAM Address Multiplexer consists of one integrated circuit, U41. This device takes 14 RAM address bits (RAM0 through RAM13) and selects either the top 7 bits or the lower 7 bits, depending on the state of ROW-1/COL-0. The halves of the address are sent to the Graphics Memory Board sequentially during a RAM cycle.

U41 also contains a 7 bit counter which generates the refresh address during refresh cycles. This counter is both clocked and enabled by the REFRESH-1 signal. When REFRESH-1 goes true, the 7 bit output of the counter is used as a refresh address to the Graphics RAM. At the end of the cycle (negative-going edge of REFRESH-1) the counter is incremented by one.

Bank Select Logic

Refer to Schematic 3-4 and Figure 5-5. The Bank Select Logic looks at the top two RAM address bits, RAM14 and RAM15 and enables one of four gates. These gates produce RAS0EN-1 through RAS3EN-1 (RAM Address Select 0 Enable, etc.) when the signal RAMCYCLE-1 is also true. The RAM Address Select signals select one of the four banks of RAM. When REFRESH-0 is true, all four of the outputs go high. Thus, all four banks of RAMs are enabled during a refresh cycle.

Write Enable Logic

Refer to Schematic 3-4 and Figure 5-5. The purpose of the Write Enable Logic is to enable writing into Graphics RAM during Processor input cycles. When I/O RAMS-0 and W-1/R-O are both high, the circuit produces WRTSTB-1 (Write Strobe). When TP160 (TFILL-0) is grounded, all cycles become writes.

Column Address Strobe Enable (CAS Enable)

Refer to Schematic 3-4 and Figure 5-5. The CAS Enable circuit produces RAMCYCLE-1. RAMCYCLE-1 is the OR function of IORAMS-0 and DCRAMS-0. It goes to the Graphics Memory Board and the Bank Select Logic.

Power Supply Sync Circuit

Refer to Schematic 3-4. The power supply sync circuitry takes the 18-MHz, FCLK-1 signal and divides it by 390. It produces a narrow negative-going output pulse, eight FCLK-1 cycles wide. This pulse, inverted, is called PSYNC-1. PSYNC-1 serves to synchronize the power supply's switching frequency with the display to prevent a "ripple" effect due to power supply noise. If this signal becomes active too soon after the power supply starts to turn on, the power supply ends up in a permanent current limiting state, and never turns on. Q373, C375, R375, CR379,and CR380 delay the activation of PSYNC until 5V has been up for about 750 ms.

GRAPHICS MEMORY BOARD

The Graphics Memory Board, shown in Schematics 4-1 through 4-4, contains random access memory which is used to store bit patterns for graphics characters. The memory array is organized into four banks which are each 16384 (16K) words long. The words are 24 bits long; however, they are divided into three 8-bit sections termed Planes A, B, and C. The standard complement of memory is 16K and this is installed in Bank 3 (high order 16K of address space). The array is expanded in 16K (one bank) increments, from the top downward, as options are added (see Figure 4-7).

Except for a few gates and a data bus driver, the Graphics Memory board is populated by 16K by 1-bit dynamic RAM devices. All addressing and control functions for these RAMs are performed by the Graphics Memory Controller board. The Graphics Memory derives only power from the Mother board.

During a memory cycle, the RAMs receive a 14-bit, multiplexed address from the Graphics Memory Controller. The address is sent as two sequential 7-bit halves over address lines MUX0 through MUX6. First, The Ram Address Multiplexer (on the Graphics Memory Controller) sends a 7-bit "row address". The row address is latched into a register within the RAM devices. Next, a 7-bit "column address" is sent and latched into a different register within the RAMs. About 200 ns after the column address has been latched, the data at the selected address appears at the RAMs' outputs. See Figure 4-12, which depicts the waveforms for a write cycle.

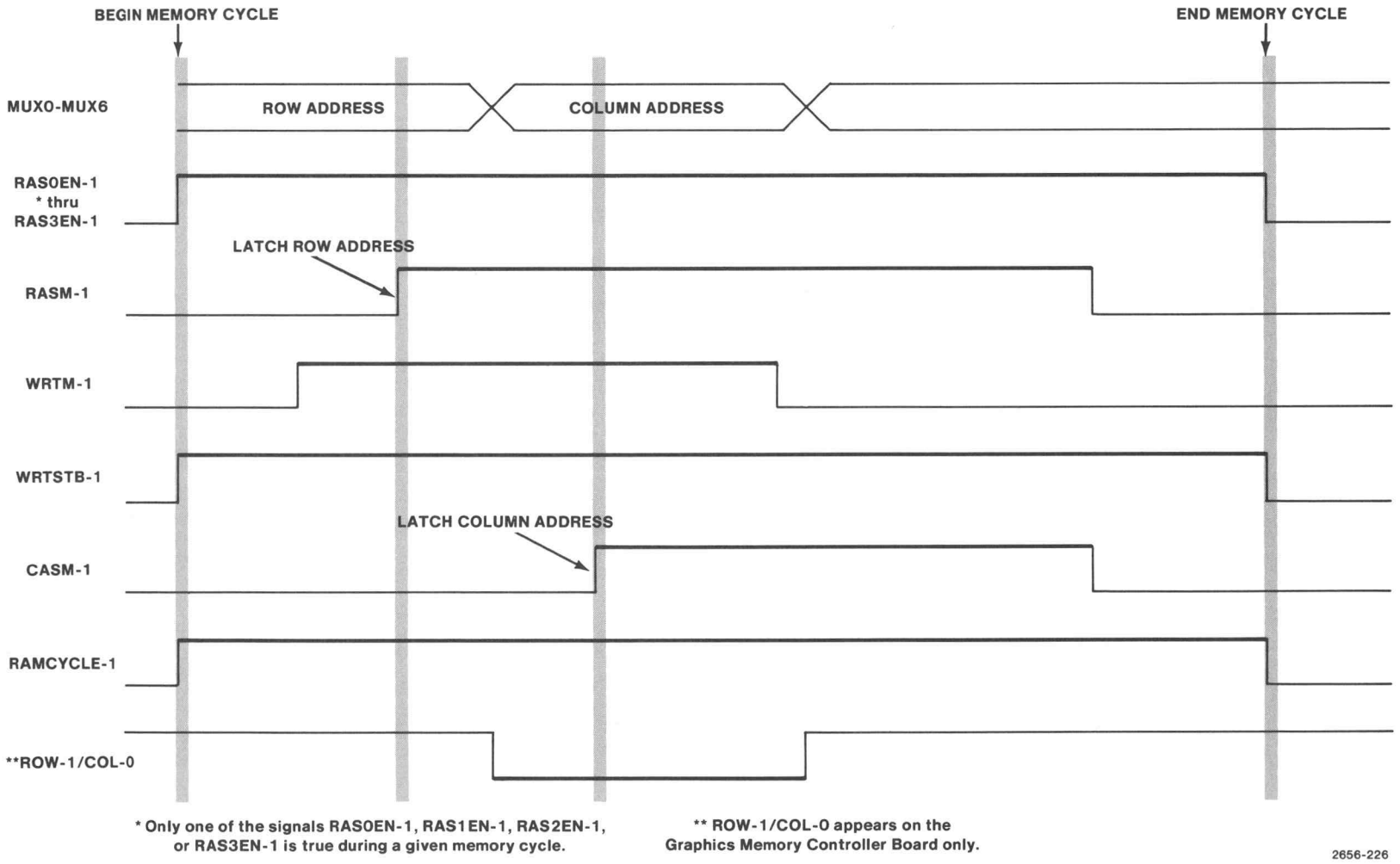


Figure 4-12. Graphics Memory Write Cycle Waveforms.

PROCESSOR BOARD

Introduction

The Processor Board controls the terminal. This control takes place through two main sections:

- The Control and Address Section.
- The Memory Section.

The control and address section contains:

- The Microprocessor.
- The Memory Decode.
- The Interrupt Control.
- The Bus Drivers.
- The Wait State Logic.

The Control and Address Section executes the firmware and processes the terminal data (see Figure 4-13).

The Memory Section contains the firmware EPROM's, scratch RAM, and battery backed up CMOS RAM (see Figure 4-14). The memory section holds the system firmware, system variables, and system parameters.

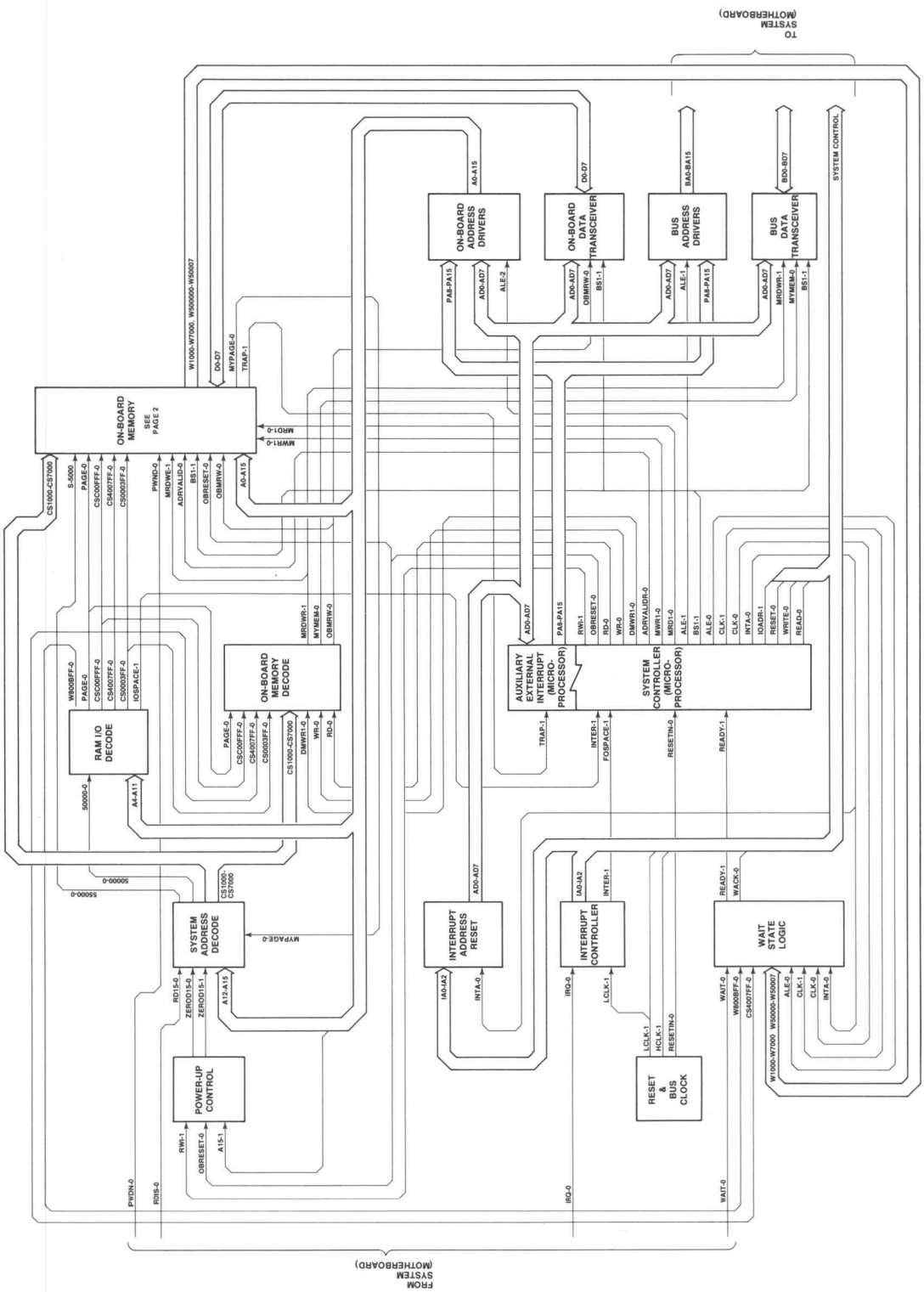


Figure 4-13. 4027A Processor Board (Control and Address) Block Diagram.

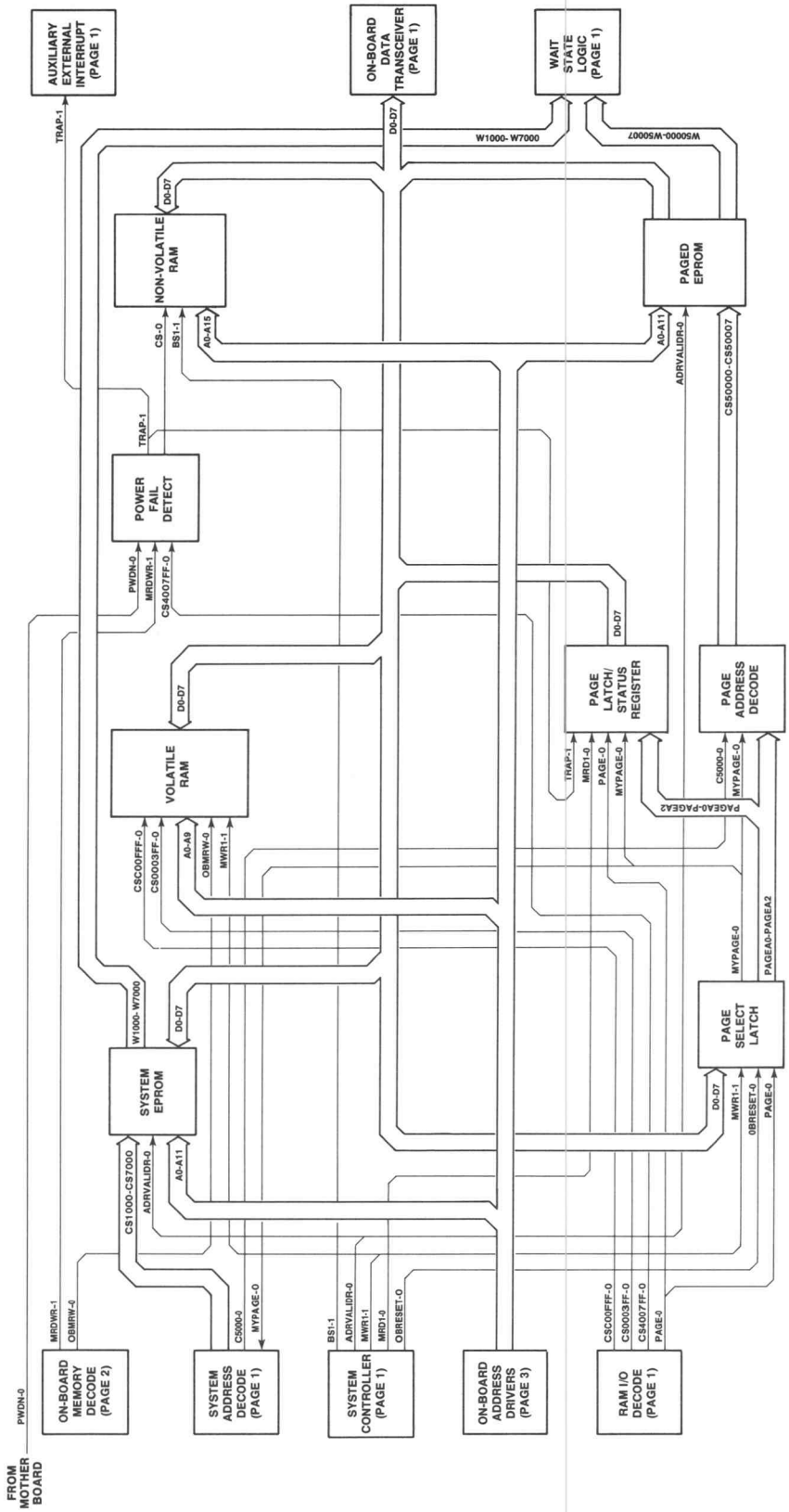


Figure 4-14. 4027A Processor Board (Memory) Block Diagram.

Control and Address Section

The Control and Address Section of the Processor board contains the following circuitry:

- System Controller (part of microprocessor)
- Auxilary External Interrupt (part of microprocessor)
- Reset and Bus Clock
- Power-up Control
- System Address Decode
- RAM I/O Decode
- On-board Memory Decode
- On-board Address Drivers
- On-board Data Transceiver
- Bus Address Drivers
- Bus Data Transceiver
- Interrupt Controller
- Interrupt Address Reset
- Wait State Logic

System Controller (Part of Microprocessor)

The microprocessor has been divided into two functional parts, the System Controller and the Auxiliary External Interrupt. The System Controller provides the control lines for the terminal and monitors the reset, system interrupts, and system ready.

Input Line Definitions

INTER-1. The INTER-1 line is used to inform the microprocessor that an interrupt is ready to be processed.

READY-1. The READY-1 line is used to hold the microprocessor in a wait state, during a read or write operation, until the data is ready for the microprocessor or stored into the addressed device.

RESETIN-0. The RESETIN-0 line is low during power-up or when the reset button is pressed.

Output Line Definitions

ADRVALIDR-0. The ADRVALIDR-0 (Address Valid Read) line is used for EPROM output control during a EPROM read cycle.

ALE-1. The ALE-1 (Address Latch Enable) line is used to latch the multiplexed address/data lines (AD0-AD7) into the address latches (A0-A7 and BA0-BA7).

ALE-0. The ALE-0 line is the inverse of the ALE-1 line and is used to reset the Wait State Logic before a memory operation.

BS1-1. The BS1-1 (Buffered Status 1) line, when high, indicates a memory read, opcode fetch, or an interrupt acknowledge.

CLK-1. The CLK-1 (Clock) line is a 5 Mhz clock used as a timing signal to the Wait State Logic.

CLK-0. The CLK-0 line is the inverse of the CLK-1 line also used by the Wait State Logic.

DMWR1-0. The DMWR1-0 (Delayed Memory Write) line is the same as the MWR1-0 line but delayed on both rising and falling edges.

INTA-0. The INTA-0 (Interrupt Acknowledge) line is used instead of the RD-0 line during the instruction cycle after the INTER-1 (Interrupt Request) line is high and accepted by the microprocessor.

IOADR-1. The IOADR-1 (Input/Output Address) line is used for access to the I/O space addresses (X'0800' to X'0BFF').

MRD1-0. A low level on the MRD1-0 (Memory Read) line indicates the selected memory device is to be read and the data bus is available for data transfer.

MWR1-0. A low level on the MWR1-0 (Memory Write) line indicates the data on the data bus is to be written into the selected memory location.

OBRESET-0. The OBRESET-0 (On Board Reset) line indicates that the microprocessor is being reset and is used on the processor board to reset the Page Select Latch and forces the Power-up Control to output an address of X'1000' on the address bus.

RD-0. A low level on the RD-0 (Read) line indicates to the selected memory or I/O device is to be read and that the data bus is available for data transfer.

READ-0. The READ-0 line is the same as the MRD1-0 line and is used off the processor board.

RESET-0. The RESET-0 line is the off board reset signal for the system.

RWI-1. The RWI-1 (Read Write Interrupt Acknowledge) line is high when there is a memory access.

WR-0. A low level on the WR-0 (Write) line indicates the data on the data bus is to be written into the selected memory or I/O location.

WRITE-0. The WRITE-0 line is the same as the MWR1-0 line used off the processor board.

Auxiliary External Interrupt (Part of Microprocessor)

The Auxiliary External Interrupt is part of the microprocessor along with the System Controller. The Auxiliary External Interrupt has the address and data lines, and the TRAP-0 interrupt input.

Line Definitions

AD0-AD7. The AD0-AD7 (Address Data) lines are the multiplexed address/data lines. On the falling edge of ALE-1 the AD0-AD7 lines contain the low 8 bits of address and are latched into the On-board Address Drivers and the Bus Address Drivers. During a low level on the RD-0, WR-0, or INTA-0 lines, the AD0-AD7 lines contain the 8 bits of data.

PA8-PA15. The PA8-PA15 (Processor Address) lines are the most significant 8 bits of the memory address.

TRAP-1. The TRAP-1 line is a non-maskable reset interrupt. The TRAP interrupt is recognized at the same time as an INTR-1 interrupt. The TRAP interrupt is used during power down to disable access by the microprocessor to the non-volatile RAM.

Reset and Bus Clock

The Reset and Bus Clock supplies the RESETIN-0 to the microprocessor and the off board timing clocks.

Output Line Definitions

HCLK-1. The HCLK-1 (High Frequency Clock) line is a 18.432 MHz clock, used off the processor board.

LCLK-1. The LCLK-1 (Low Frequency Clock) is a 2.048 MHz clock used by the Interrupt Controller and off the processor board.

RESETIN-0. The RESETIN-0 line supplies the power-up or master reset to the microprocessor.

Power-Up Control

Figure 4-15 shows the processor board address space. Note that the system firmware starts at X'1000' and that system RAM occupies addresses from X'0000' to X'03FF'. The system RAM contains the eight interrupt vectors (pointers to interrupt handling subroutines) at addresses X'0000', X'0008', X'0010', etc.

On power-up, or when the master reset is pressed, the microprocessor always branches to address X'0000'. The Power-up Control modifies the address to X'1000' by forcing the System Address Decode to disable the X'0000' to X'0FFF' address space and enabling the X'1000' to X'1FFF' address space. The address modify is removed by a memory operation in the X'8000' to X'FFFF' address space (address bit A15 high).

Line Definition

A15-1. The A15-1 (Address) line is used with RWI-1 to clear the forced X'1000' address.

OBRESET-0. The OBRESET-0 (On Board Reset) line is used to force the X'1000' address at power-up or master reset.

RWI-1. The RWI-1 (Read Write Interrupt Acknowledge) line is used with the A15-1 line to clear the forced X'1000' address.

ZERODIS-0. The ZERODIS-0 (Zero Disable) line is used to enable the X'1000' address space.

ZERODIS-1. The ZERODIS-1 (Zero Disable) line is used to disable the X'0000' address space.

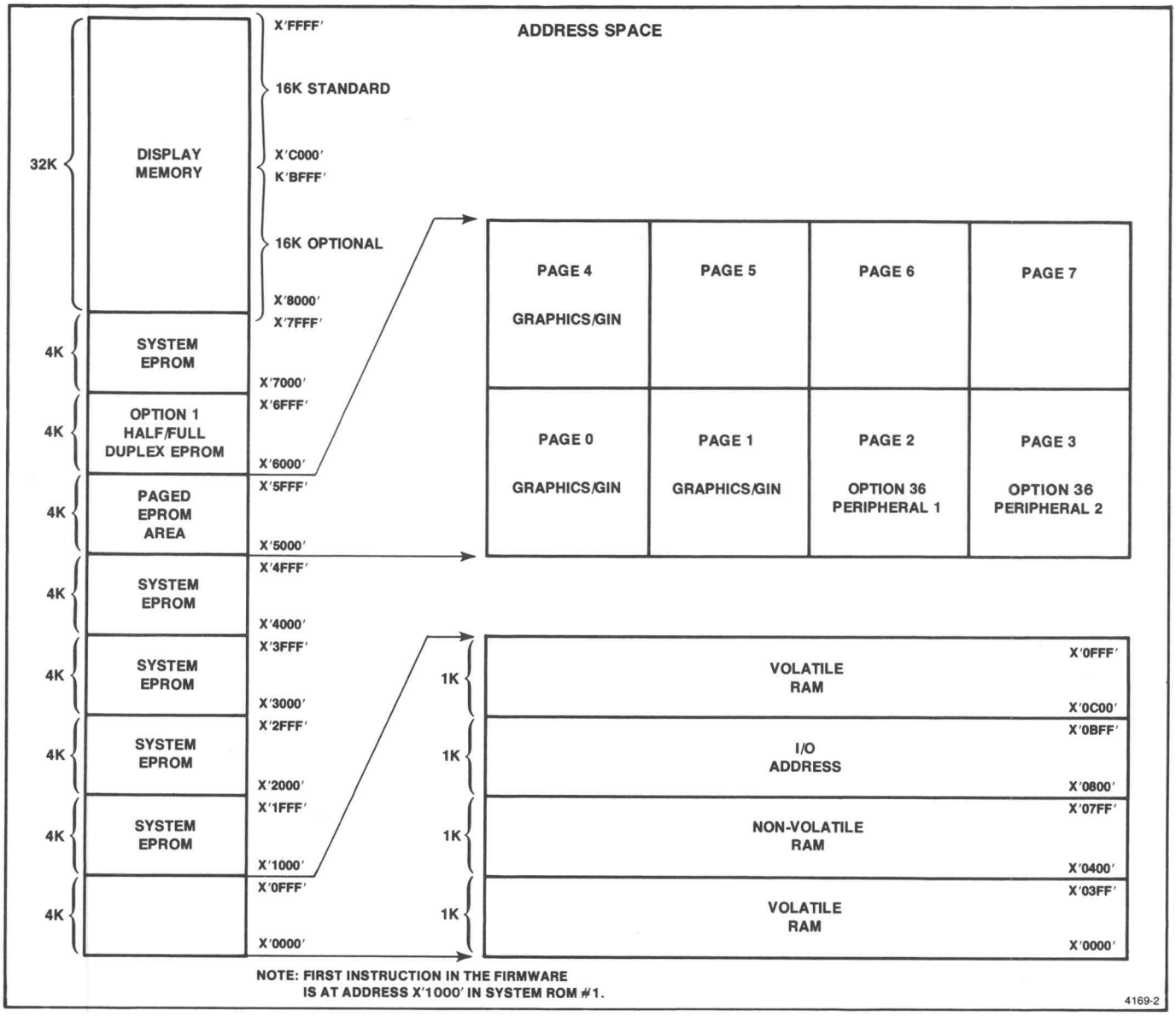


Figure 4-15. Processor Board Address Space.

System Address Decode

The System Address Decode divides the lower 32K bytes of memory into eight 4K sections (refer to Figure 4-15). Three of the 4K sections can be modified, either enabled (X'1000' to X'1FFF' section during power-up/reset) or disabled (X'5000' to X'5FFF' section and X'6000' to X'6FFF' section).

Line Definitions

A12-A14. The A12-A14 (Address) lines are decoded into eight 4K sections (X'0000' to X'7000').

A15-1. The A15-1 (Address) line address being low with ZERODIS-1 line low enables the System Address Decode.

CS1000-CS7000. The CS1000-CS7000 (Chip Select) lines are the 4K section decode lines.

MYPAGE-0. The MYPAGE-0 line when low is used to enable the X'5000' to X'5FFF' section on the processor board. When MYPAGE-0 is high, the X'5000' to X'5FFF' section is off the processor board.

RDIS-0. The RDIS-0 (ROM Disable) line is used to disable the X'6000' to X'6FFF' section and allow the program memory to be off the processor board.

S0000-0. The S0000-0 (Select) line is used by the RAM I/O Decode (X'0000' to X'0FFF' memory section).

S5000-0. The S5000-0 (Select) line is used by Paged Address Decode for the X'5000' to X'5FFF' section.

ZERODIS-0. The ZERODIS-0 (Zero Disable) line is used to force the CS1000-0 line to be active when X'0000' is modified to X'1000' during power-up or master reset.

ZERODIS-1. The ZERODIS-1 (Zero Disable) line is used to disable the System Address Decoder during power-up or master reset.

RAM I/O Decode

The RAM I/O Decode is used to section the X'0000' to X'0FFF' space into 1K sections and to decode the X'0800' address of the Page Select Latch and the Page Latch/Status Register (refer to Figure 4-13).

Line Definitions

A4-A7. The A4-A7 (Address) lines are used to decode the X'0800' address (PAGE-0 line).

A8-A11. The A8-A11 (Address) lines are used to section the X'0000' to X'0FFF' address space into four 1K sections.

CS0003FF-0. The CS0003FF-0 (Chip Select) line enables one of the 1K sections of the volatile RAM (X'0000' to X'03FF').

CS4007FF-0. The CS4007FF-0 (Chip Select) line enables the non-volatile RAM (X'0400' to X'07FF').

CSC00FFF-0. The CSC00FFF-0 (Chip Select) line enables one of the 1K sections of the volatile RAM (X'0C00' to X'0FFF').

IOSPACE-1. The IOSPACE-1 (Input/Output Space) line indicates that the address is in the range of X'0800' to X'0BFF'.

PAGE-0. The PAGE-0 line is used to enable the Page Select Latch and the Page Select/Status Register (X'0800' to X'080F').

S0000-0. The S0000-0 (Select) line enables the RAM I/O Decode (X'0000' to X'0FFF').

W800BFF-0. The W800BFF-0 (Wait) line signals the Wait State Logic to wait. The W800BFF-0 line is the inverse of the IOSPACE-1 line.

On-board Memory Decode

The On-board Memory Decode uses PAGE-0, CS0003FF, CS4007FF, CSC00FFF-0, and CS1000-CS7000 to indicate that the device addressed is on the processor board. The output controls the direction and output of the On-board Data Transceiver and the Bus Data Transceiver.

On-board Address Drivers

The On-board Address Drivers are used to supply the A0-A15 address lines used on the processor board. The PA8-PA15 lines from the microprocessor arealways passed through to the A8-A15 address lines. The AD0-AD7 multiplexed address/data lines supply the A0-A7 address lines. The A0-A7 address lines are latched on the falling edge of ALE-1.

On-board Data Transceiver

The On-board Data Transceiver is used to pass data between the microprocessor and the devices on the processor board. The on-board data transceiver is oly enabled when the device selected is on the processor board. The data direction is controlled by the BS1-1 line from the System Controller.

Bus Address Drivers

The Bus Address Drivers supply the BA0-BA15 address lines used off the processor board. The PA8-PA15 lines from the microprocessor are buffered to the BA8-BA15 address lines. The BA0-BA7 address lines are latched on the falling edge of ALE-1. The AD0-AD7 multiplexed address/data lines supply the BA0-BA7 address lines.

Bus Data Transceiver

The On-board Data Transceiver is used to pass data from the microprocessor to the devices off the processor board. The data direction is controlled by th BS1-1 line from the System Controller and by the On-board Memory Decode. The Bus Data only transfers data to the microprocessor during a read of a device off the processor board.

Interrupt Controller

Several parts of the terminal can request processor interrupts; the keyboard and host port on the Data Cummunications Interface Board are examples Each device capable of requesting an interrupt is assigned an interrupt value (0 to 7), usually by setting straps on the circuit board.

The eight interrupt values are numbered from 0 to 7. The interrupt value tells the microprocessor which interrupt handling subroutine to use in responding to an interrupt.

An address counter issues an interrupt value in binary form on the interrupt address lines IA0-IA2. Any device which is assigned the current interrupt address value may request an interrupt by pulling the interrupt request line, IRQ, low.

The Interrupt Controller circuitry examines the IRQ line after the interrupt address value is issued. If there is an interrupt request, the microprocessor is interrupted by the INTER line. Further increments of the interrupt address lines and sampling of the interrupt request line are disabled until the IRQ line goes high.

Interrupt Address Reset

After an interrupt has been acknowledged, the microprocessor requires an instruction (usually a Restart instruction, binary 11AAA111) to be placed on the data bus. The Interrupt Address Reset does this.

When an interrupt is acknowledged, the INTA-0 signal enables the tri-state buffer, placing the binary word 11AAA111 on the data bus. The value of AAA is the interrupt value from the counter in the Interrupt Controller. The data word is interpreted by the microprocessor as a RESTART instruction, causing the microprocessor to branch to the subroutine which starts at binary address 00000000 00AAA000.

Wait State Logic

The Wait State Logic is used to hold the address device until the device is ready for the data transfer. The processor board wait signals are W1000-W7000, W50000-W50007, and CS4007FF-0. The off board wait signals are WAIT-0, W8007FF-0 (I/O space), and A15-1 (display memory). The WAIT-0 line is the wait input to the processor board. When low, WAIT-0 holds the microprocessor until the external device is ready.

Memory Section

The Memory Section of the Processor board contains the following sections:

- System EPROM
- Volatile RAM
- Power Fail Detect
- Non-volatile RAM
- Page Select Latch
- Page Latch/Status Register
- Page Address Decode
- Paged EPROM

For the block diagram of the processor memory refer to Figure 5-2. Refer to Figure 5-3 for the processor board address space.

System EPROM

The System EPROM is made up of six 4K EPROM locations. The base address of the System EPROM locations are X'1000', X'2000', X'3000', X'4000', X'6000', and X'7000'. The EPROMs are selected by CS1000-0, CS2000-0, CS3000-0, CS4000-0, CS6000-0, and CS7000-0. The output of the EPROMs is controlled by the ADRVALID-0 line. The W1000-W7000 lines are used to delay the microprocessor, if slow EPROMs are used. The data is transferred out on the D0-D7 lines.

Volatile RAM

The Volatile RAM consist of two 1K RAM areas. The base addresses of the Volatile RAM are X'0000' and X'0C00'. The RAMs are selected by CS0003FF-0 and CSC00FFF-0 with OBMRW-0 (On-Board Memory Read or Write). The write control line used by the RAMs is the MWR-0 line. Data is transferred in and out through the D0-D7 lines.

Power Fail Detect

The Power Fail Detect circuitry protects the data in the CMOS (non-volatile RAM) during power-off, power-on sequences. It does this by disabling the CMOS RAM and sending a TRAP-1 interrupt to the microprocessor. During an intermittent power down cycle, the system firmware continues to execute when TRAP-1 is lowered.

The RAM is disabled and the TRAP-1 signal sent when the Power Supply sends a PWDN-0 signal to warn of impending loss of power on the +5 V line. On power-up, the RAM is not enabled, and the TRAP signal is high, until the terminal is completely powered-up. This prevents transients on the address lines from garbling the data in the CMOS RAM.

Non-Volatile RAM

The non-volatile RAM holds settings such as baud rate, parity, etc., when the terminal is turned off. The CMOS RAM receives power from the battery BT1001 through the charging circuits. The RAM is disabled and the RAM address and data lines are forced low when the terminal is turned off to conserve the battery.

Two 1K x 4 CMOS RAMs are used. One RAM holds the least significant four bits (D0-D3) of each 8-bit word; the other holds the most significant four bits (D4-D7).

Page Select Latch

The Page Select Latch is used to select one of eight pages in the X'5000' to X'5FFF' address space. The pages are numbered 0 to 7 and are addressedin binary by the PAGE0-PAGE2 lines. When low, the MYPAGE-0 line is used to enable the X'5000' address space on the processor board. The address of the Page Select Latch is at X'0800' and selected by the PAGE-0 line.

The Page Select Latch also has four outputs that can be used as test points.

Page Latch/Status Register

The Page Latch/Status Register is used to read the Page Select Latch, the status of the TRAP-1 interrupt line, and a set of three test flags. The Page Latch/Status Register is addressed at X'0800' and selected by the PAGE-0 line.

The test flags (J757, J758, and J759) are used for testing and flagging to the firmware. Table 4-5 describes the designated uses.

Table 4-5

TEST FLAG CODING

J759	J758	J757	Function
IN	IN	IN	Not Used
IN	IN	OUT	Not Used
IN	OUT	IN	Not Used
IN	OUT	OUT	4025A Opt. 38 Installed
OUT	IN	IN	Not Used
OUT	IN	OUT	Power Up, Scope Test Loop
OUT	OUT	IN	Cycle Room Test
OUT	OUT	OUT	Normal Operation

NOTE

"NOT USED" defaults to "NORMAL OPERATION".

Page Address Decode

The Page Address Decode is used to select the Paged EPROM. The S5000-0 and MYPAGE-0 lines select the decoder. The PAGE0-PAGE2 lines select which page is to be active. The CS50000-CS50007 lines are the chip enable to the Paged EPROM.

Paged EPROM

The Paged EPROM is made up of eight pages of 4K each starting at address X'5000'. The select lines are the CS50000-CS50007 lines. The output enable is controlled by the ADRVALID-0 line. The W50000-W50007 lines are used to delay the microprocessor, if there are slow EPROMS used. The data is transferred over the D0-D7 lines.

DELUXE COMMUNICATIONS BOARD

The Deluxe Communications Board provides two data communications ports: the Host Port and the Keyboard Port. These ports are interfaces through which the microprocessor communicates with the host computer and the keyboard. Each port appears to the microprocessor to be a series of words in its I/O address space.

The circuitry on the Deluxe Communications Board may be grouped into three categories. These are the:

- Processor Interface. Included in this category are circuits which interface the Host and Keyboard Ports with the terminal's microprocessor.
- Keyboard Port, which interfaces the microprocessor with the the keyboard.
- Host Port, which interfaces the microprocessor with the host computer.

There are three block diagrams covering this circuitry: Figure 5-7 is an overall diagram, Figures 5-8 and 5-9 cover the Keyboard and Host Ports, respectively.

Processor Interface Circuitry

I/O Registers

The microprocessor can control signal lines on the Deluxe Communications Board by writing into several I/O registers and it can read the states of signal lines by reading from these I/O registers.

Each I/O register consists of two devices: the "read" and "write" halves of the register. The read half of an I/O register usually consists of tri-state buffers which, during read operations, place the data on the terminal's data bus. The write half of an I/O register is usually a latch which, during write operations, saves the data on the terminal's data bus. The latch outputs drive signal lines on the Deluxe Communications Board.

Tables 4-6 and 4-7 list the Host and Keyboard Port I/O registers along with their functions and I/O addresses.

Table 4-6

HOST PORT I/O REGISTERS

Register Name	**Address**	**Function** (W=write, R=read)
Host Port Status Word	X'0820'	Bit 0(W): OUTRDY Interrupt Enable Bit 1(R): OUTRDY (Output Ready) Bit 2(W): INRDY Interrupt Enable Bit 3(R): INRDY (Input Ready) Bit 4(W): LSC Interrupt Enable Bit 5(R): LSC (Line Status Change) Bit 6(R): TXE (Transmitter Empty) BIT 7(R): Interrupt Flag
Line Protocol Word	X'0821'	Bit 0(W): LP0 Bit 1(W): LP1 Bits 2-7: Unused
Line Status Word	X'0822'	Bit 0(R): DSR (Data Set Ready) Bit 1(R): SDCD (Sec. Data Carrier Detect) Bit 2(R): DCD (Data Carrier Detect) Bit 3(R): CTS (Clear to Send) Bit 4(W): DTR (Data Terminal Ready) Bit 5(W): SRTS (Sec. Req. to Send, pin 11) Bit 6(W): SRTS (pin 19) Bit 7(W): RTS (Request to Send)
Baud Rate Word	X'0823'	Bits 0-3(W): Receive Baud Rate Bits 4-7(W): Transmit Baud rate
USART Data Word	X'0824' X'0825'	Bits 0-7: Read Data Bits 0-7: Write Data
USART Programming and Status Words	X'0826' and X'0827'	Read: (USART Status Word) Bit 0: TXRDY (Transmitter Ready) Bit 1: RXRDY (Receiver Ready) Bit 2: TXE (Transmit Enable) Bit 3: PE (Parity Error) Bit 4: OE (Overflow Error) Bit 5: FE (Framing Error) Bit 6: Sync Detect Bit 7: Ring Indicator
		Write: (USART Programming) Bits 0-7: (See USART Circuit Description.)

Table 4-7

4027A KEYBOARD PORT I/O REGISTERS

Register Name	Address	Function (Wwrite, Rread)
Keyboard Port Status Word	X'0830'	Bits 0-1: Unused Bit 2(W): KEYRDYIE (KEYRDY Interrupt Enable) Bit 3(R): KEYRDY (Keyboard Data Ready) Bits 4-6: Unused Bit 7(R): IFLAG (Interrupt Flag)
Keyboard Reset	X'0821';	A write to this word, regardless of the data written, resets the Keyboard Port.
Keyboard Data Word	X'0822'	Bits 0-7(R): Code showing which keys are currently being pressed. Bit 0(W): COMMAND LOCKOUT light. Bit 1(W): NUMERIC LOCK light. Bit 2(W): TTY LOCK light. Bit 3(W): INSERT MODE light. Bits 4-6: Unused. Bit 7(W): Bell.

Data Buffers

(Schematic 6-1)

The Data Buffers buffer the data on the terminal's data bus BD0-BD7, placing it on a local data bus (lines D0-D7) within the Communications Interface Board. During reads from I/O registers on the board, the data buffers buffer the lines D0-D7, placing their signals on the terminal data bus.

The 74LS243 bidirectional Data Buffers are steered by the MYRDADR (My Read Address) signal from the Address Decoders. When MYRDADR is false, the terminal data bus BD0-BD7 drives the local data lines D0-D7. When MYRDADR is true, the microprocessor is reading from the I/O register on the Deluxe Communications Board; in that case, the local data lines D0-D7 drive the terminal data bus lines BD0-BD7.

Address Decoders

(Schematic 6-1)

The Address Decoders monitor the terminal's address bus (BA0-BA9) and the IOADR, READ, and WRITE lines. The decoder outputs enable the I/O registers in the Keyboard and Host Ports.

The Address Decoder outputs are:

- **MYRDADR (My Read Address).** Signals a read from an address on the Deluxe Communications Board (an address in the ranges X'0820'-X'0827' and X'0830'-X'0837').
- **KDR (Keyboard Data Read).** True during a read from X'0832'.
- **KDW (Keyboard Data Write).** True during a write to X'0832'.
- **KRST (Keyboard Reset).** True during a write to X'0831'.
- **KSWR (Keyboard Status Word Read).** Signals a read from X'0830'.
- **KSWW (Keyboard Status Word Write.** Signals a write to X'0830'.
- **HCE (Host Port Chip Enable).** Enables the USART during reads or writes at addresses X'0824'-X'0827'.
- **HBRW (Host Baud Rate Word Write).** Signals a write to X'0823'.
- **HLSR (Host Line Status Word Read).** Signals a read from X'0822'.
- **HLSW (Host Line Status Word Write).** Signals a write to X'0822'.
- **HLPW (Host Line Protocol Word Write).** Signals a write to X'0821'.
- **HSWR (Host Port Status Word Read).** Signals a read from X'0820'.
- **HSWW (Host Port Status Word Write).** Signals a write to X'0820'.

Interrupt Address Decoders

(Schematic 6-2)

The Interrupt Address Decoders monitor the terminal's IA0-IA2 interrupt address lines and the HIRQST (Host Port Interrupt Request) and KIRQST (Keyboard Port Interrupt Request) signals. When the IA0-IA2 lines are at the Host Port's interrupt address and the HIRQST signal is true, the Interrupt Address Decoder sends the IRQ (Interrupt Request) signal to the Processor. Likewise, IRQ is sent when KIRQST is true and the Keyboard Port's interrupt address is on the IA0-IA2 lines.

Host Port

The description of the Host Port covers three major topics:

- First, there is an overview of data communications. This includes asynchronous and synchronous modes, the RS-232 interface, full and half duplex modes, remote and local echo.
- Next, the communications firmware is discussed: how the Host Port is initialized at power-up, how characters are transmitted, how characters are received.
- Finally, the actual circuitry in the Host Port is discussed.

Data Communications Overview

There are two main methods of serial data communications: the **synchronous** and **asynchronous** modes. The 4027A uses asynchronous mode.

In asynchronous data communications, each character is framed with **start** and **stop** bits (Figure 4-16). When no characters are being sent, the terminal holds the TDATA (Transmit Data) RS-232 line low, in the "mark" or binary 1 condition. Each character begins with a **start bit**, always binary 0 (RS-232 high or "space" condition). Following the start bit are several **selector bits** which determine which character is being sent. (If the ASCII code is used, there are usually eight selector bits: seven bits for the ASCII character — least significant bit first — followed by a parity bit.) Each character ends with a **stop bit**, always binary 1 (RS-232 low or "mark" condition).

The start bit signals the receiving terminal that a character is beginning. The stop bit (or bits) gives the terminal time to recover before the next character comes along. Using the start bit, the receiver re-synchronizes itself at the start of each character. Thus, the transmitting and receiving terminals can have slightly different data rates and still be able to communicate.

In synchronous data communications, the start and stop bits are omitted. The first bit of one character immediately follows the last bit of the preceding character (Figure 4-16).

NOTE

The 4027A system firmware does not support synchronous data communications. However, there is circuitry on the Deluxe Communications Board for use with synchronous mode, should firmware ever be provided for it.

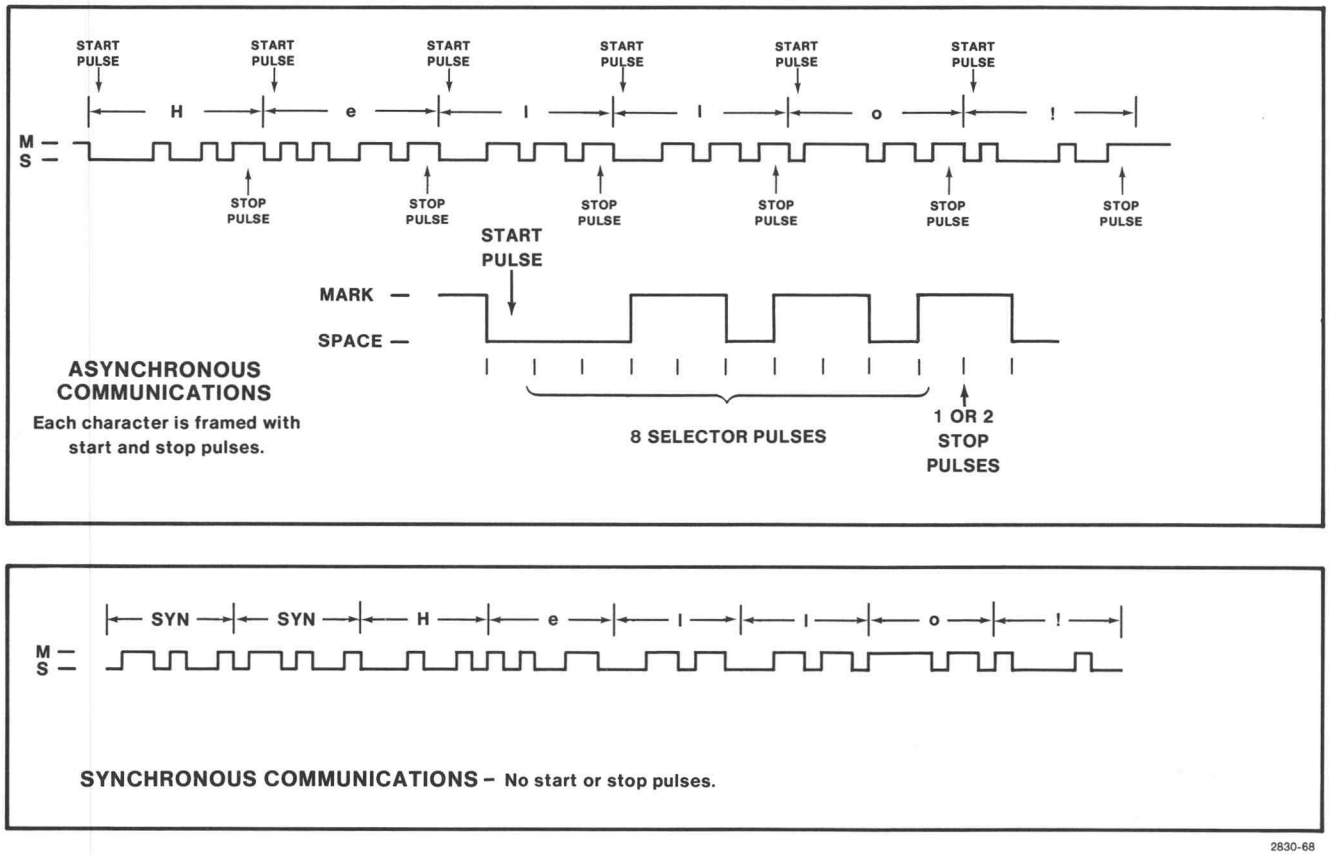


Figure 4-16. Asynchronous and Synchronous Serial Data Formats.

The signals between a terminal and its modem (or other data communications equipment) are defined in Electronic Industries Association Standard RS-232-C. The voltages at the terminal-modem connector may be between +3 and +25 volts (an "RS-232 high"), or between —3 and —25 volts (an "RS-232 low"). Zero volts is commonly interpreted as an RS-232 low. The TDATA and RDATA (Transmitted and Received Data) signals have a low defined as "mark" or binary one, and a high defined as "space" or binary zero. For the other RS-232 signals, a "true" or binary one is a high, and a "false" or zero is low.

The more important RS-232 signals are: TDATA (Transmitted Data), RDATA (Received Data), DSR (Data Set Ready), DTR (Data Terminal Ready), RTS (Request to Send), CTS (Clear to Send), DCD (Data Carrier Detect), SRTS (Secondary Request to Send), SDCD (Secondary Data Carrier Detect), XMT CIOCK, and RCV CLOCK. Usually not all of these signals are used with any particular modem.

- In ordinary asynchronous full duplex communications, the TDATA and RDATA signals are always used. The following may also be used: DSR, DTR, RTS, DCD.

- In asynchronous half duplex communications (with "supervisory secondary carrier"), the following additional signals are used: SRTS, SDCD.

Every 4027A is equipped with the firmware for **full duplex** data communications. In this communications mode, terminal and modem can send and receive simultaneously. This mode is often used in computer time-sharing systems, especially at data rates of 300 baud or less. Commonly **full duplex remote echo** is used, in which the computer echoes each character sent by the terminal and the echo, rather than the transmitted character, is displayed on the terminal's screen.

However, full duplex communications requires that both "transmit" and "receive" data channels be available. At higher data rates, the simpler types of modems cannot fit both data channels into the same telephone line. One way to overcome this is to use separate telephone lines; another is to use a **half duplex** communication mode.

When equipped with Option 01, the 4027A offers two half duplex communication modes: **half duplex normal** and **half duplex with supervisor.**

In half duplex normal mode, the computer and the terminal take turns using the data channel. Normally, the terminal is operating in buffered mode. As the 4027A transmits to the computer, it ends each line with an **end-of-line string** (previously defined by an EOL command). The end-of-line string serves as a **line turnaround character** (or characters); it tells the computer that it may transmit to the 4027A. The computer responds by sending some text, which must end with a **prompt string** (previously defined with the PROMPT command). The prompt string serves to turn the line around again: it tells the 4027A that it may send the next line of text.

To avoid the need for line turnaround characters, often half duplex with supervisor mode is used. In this mode, the computer controls the direction of data transfer. To do this, it uses a secondary "supervisory" carrier tone sent over the telephone line. (Sending the SRTS signal to a modem causes the modem to place this secondary carrier on the telephone line.) The process is as follows:

1. Except when sending data to the terminal, the computer listens on the telephone line and sends a secondary carrier tone to tell the terminal that it is listening. (Actually, the computer asserts SRTS and its modem sends the secondary carrier).

   When the 4027A's modem hears the secondary carrier, it asserts SDCD (Secondary Data Carrier Detect). So long as SDCD is true, the 4027A stays in transmit state and asserts RTS (Request to Send), causing its modem to send the primary carrier tone. Even if it has nothing to say, the 4027A must stay in transmit state and send RTS until the computer commands it to enter receive state.

2. When the computer has something to say to the terminal, it turns off the secondary carrier.

   As the secondary carrier vanishes, the 4027A's modem drops SDCD (Secondary Data Carrier Detect). The 4027A finishes sending the characters in its USART and then exits transmit mode. As it does so, it sends SRTS (Secondary Request To Send); this causes its modem to send a secondary carrier tone back to the computer. (The secondary carrier's presence means, "O.K., I'm listening.")

   Within a short time, the computer asserts RTS (Request to Send), and its modem sends the carrier tone to the terminal's modem. The terminal's modem sends DCD (Data Carrier Detect) to the terminal, and the terminal enters "receive" mode. At this point, the computer may start sending data to the terminal.

3. The 4027A remains in receive state until the computer is done sending and turns off the data carrier. When the data carrier is no longer received, the 4027A's modem turns off DCD. The 4027A responds by exiting receive mode, turning off SRTS, and turning on RTS to send a carrier tone back to the computer.

4. After a delay (to give the computer time to bring up its secondary carrier) the 4027A enters transmit mode once again and is free to send data to the computer.

Note that the computer controls the direction of data transfer. When the computer places the secondary carrier on the line, the 4027A is required to enter transmit state. Likewise, when the computer turns off the secondary carrier, the 4027A must enter receive state.

If the computer provides **no** carrier tone (neither the primary carrier nor the secondary carrier), the 4027A alternates (or "hunts") between transmit and receive states. While in transmit state, the absence of the secondary carrier forces the terminal into receive state. While in receive state, the absence of the primary carrier forces the terminal back into transmit state.

While the computer is sending to the 4027A the 4027A's operator may request transmit data by pressing the BREAK key twice in quick succession. This sends a "break" signal to the computer. (The 4027A drops SRTS, causing its modem to stop sending the secondary carrier. The computer interprets the lack of secondary carrier as a "break.") However, the computer need not honor the break request; it may just keep sending.

Most time-sharing systems use **full duplex, remote echo** mode, especially at data rates of 300 baud or less. "Full duplex" means that the terminal simultaneously transmits and receives. "Remote echo" means that as the terminal transmits, the computer echoes each transmitted character back to the terminal. It is the echoed characters, not the originally transmitted ones, which are displayed on the screen.

When full duplex communications cannot be used (as at high baud rates over ordinary telephone lines), the remote echo technique must be abandoned. In these cases, the terminal provides its own **local echo** of each transmitted character. Circuitry is provided in the Host Port to generate this local echo.

Host Port Firmware

The following descriptions of the firmware used to drive the Host Port are general in nature; the system firmware programs are not described exactly. However, the general descriptions of the firmware should help you understand how the Host Port circuitry is used.

Initializing the Host Port. On power up, or when RESET is pressed, the firmware initializes the Host Port. The initialization procedure varies slightly according to the mode of data communications to be used. If full duplex, remote echo mode is to be used, the procedure is as follows:

1. The Processor writes into the Line Protocol Word (address X'0821'), to establish which line protocol will be used. For full duplex, remote echo mode, X'00' is written at this location.

2. The Processor sets the RTS and DRT bits in the Line Status Word by writing at address X'0822'.

3. The Processor writes the baud rates into the Baud Rate Word, address X'0823'. For instance, if the terminal is to transmit and receive at 300 baud, it writes X'DD' into this word.

4. The Processor writes X'00' three times into the USART Command Word, address X'0827'. It then waits 32 microseconds, and writes X'40' into that word. Regardless of the state of the USART on power-up, this procedure is guaranteed to reset the USART, preparing it to receive a command instruction word. The Processor then waits 16 microseconds to give the USART time to reset itself.

5. The Processor writes a **mode instruction word** at address X'0827'. This word tells the USART how many bits per character to expect, what the parity will be, etc. After writing this mode instruction word, the Processor waits 32 microseconds to give the USART time to set itself accordingly.

6. The Processor then writes a **command instruction word** at address X'0827'. This word tells the USART whether to start receiving or transmitting characters. (Subsequent words written at X'0827' will be treated by the USART as command instructions. With these command instructions the Processor controls the USART, telling it to start or stop sending or receiving, or to reset itself to prepare for another mode instruction word. The operation of the USART is described in more detail later in this Section.)

7. The Processor reads from the USART Data Word (address X'0825') to clear any garbage bits that may be in the USART input buffer.

8. The Processor writes X'15' into the Host Port Status Word (address X'0820'). This enables the Host Port interrupts.

Receiving Characters. When the terminal receives characters from the host computer, the process is as follows:

1. The input interrupts are enabled on initialization, and are always enabled.

2. When the USART receives a character from the host, the Host Port requests an interrupt by pulling the IRQ line (on the Mother Board) low.

3. The Processor interrupts its current task and reads the character just received. (It reads this character from the USART Data word, address X'0824'.) It places the character in an "input queue" in another part of memory.

4. The Processor reads from the USART Status Word (address X'0826') to see if any of the USART error bits in that word have been set. If so, it places in the input queue a "flag" indicating that the character just received was in error.

5. As more characters are received, steps 2 to 4 are repeated again and again.

Later the Processor will process the input queue, doing any of the following:

- It may place the character in the display list, causing it to appear on the screen.

- If the character is part of a command to the terminal, the firmware will interpret the command and execute it.

Transmitting Characters. The following describes how characters are sent when the terminal is operating in unbuffered mode.

1. With the terminal in unbuffered mode, the **output interrupts** (the "transmitter ready" and "transmitter empty" interrupts) are always enabled. To do this, the Processor writes into the Host Port Status Word (address X'0820'), setting the appropriate bits of that word to enable these interrupts. This occurs either at initialization or when the terminal enters unbuffered mode in response to a BUFFERED NO command.

2. The Processor places the characters to be transmitted into an **output queue** or **transmit buffer**.

3. When the USART is ready to send a character, the Host Port interrupts the Processor by pulling the IRQ line (on the Mother Board) low.

4. The Processor interrupts its current task and writes the next character from the transmit buffer into the USART Data Word, address X'0825'. The Processor then returns from the interrupt handling routine and resumes its previous task.

5. The USART sends the character out the TDATA line. When it is ready to accept the next character, it generates another "transmitter ready" interrupt.

6. As long as there are characters waiting in the transmit buffer, steps 4 and 5 will repeat again and again.

7. When the transmit buffer is finally emptied, the Processor turns off the output interrupts (again by writing into the Host Port Status Word, address X'0820').

8. The next time there are characters to be transmitted, the Processor enables the interrupts again. Steps 3 through 7 are then repeated.

Host Port Circuitry

Figure 5-9 is the Host Port Block Diagram. The Host Port includes the following blocks: USART, Baud Rate Word, Baud Rate Generator, RS-232 Receivers and Transmitters, RDATA Logic, Line Status Word (Read), Line Status Word (Write), Line Protocol Word, Host Port Status Word (Read), Host Port Status Word (Write), Debouncer, LSC Detector, Host Port Interrupt Requestor, Line Protocol Logic, and Line Turnaround Logic.

USART. The USART (Universal Synchronous/Asynchronous Receiver/Transmitter) appears in Schematic 6-3 and Figure 4-17. This integrated circuit (Intel 8251A or equivalent) is especially designed for use with the microprocessor.

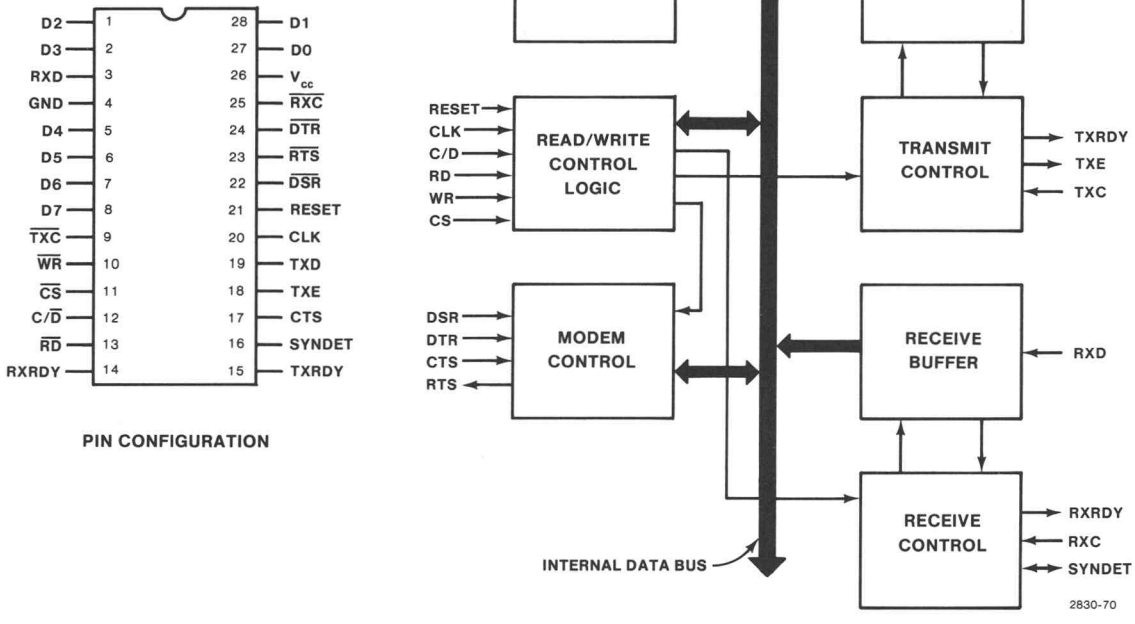


Figure 4-17. 8251A USART.

The USART is designed to occupy two different words in the microprocessor's memory. In the 4027A, address bit BA1 is applied to the USART's register select pin (C/D pin), and bits BA2-BA15 are decoded to drive the chip select pin. The least significant address bit (BA0) is ignored. The result is that each of the USART's two main registers (Command Register and the Data Register) occupy two memory addresses, and the USART as a whole occupies four memory addresses. (The Command Register occupies addresses X'0826' and X'0827', while the Data Register occupies addresses X'0824' and X'0825'.)

A read or write at address X'0824' has the same effect as a read or write at X'0825'. Likewise, reading or writing at X'0826' has the same effect as reading or writing at X'0827'.

The two main USART registers each have "read" and "write" halves, with distinct functions. Since each register also occupies two adjacent words of memory, they may be regarded as two distinct memory words.

The firmware always reads input data from address X'0824', while it always writes output data to address X'0825'. Likewise, to ascertain the USART's status, the firmware reads from X'0826', while to program the USART it writes to address X'0827'.

The USART Command and Data Registers have the following functions:

- **Input Data.** When the USART receives a character of data on its RXD pin, it stores that character in the "read" half of the USART Data Register and causes the Host Port Interrupt Requestor to request an interrupt. (The Processor is expected to respond to the interrupt by reading from the USART Data Register.)

- **Output Data.** To send a character of text out the TDATA line, the Processor writes into the "write" half of the USART Data Register.

  Once the character has been sent, the USART will cause the Host Port Interrupt Requestor to request an interrupt. When the Processor services the interrupt (by reading from the USART Status Register), it learns that the character has been sent and can then write the next character into the USART Data Register.

- **USART Status.** By reading from the USART Status Word (the "read" half of the USART Command Register), the Processor can learn whether certain events have occurred:

Bit 0: TXREADY. This bit is high (binary 1) when the USART is ready to accept another character. (Because the USART is "double buffered," it can accept another character at the same time as it is transmitting the previous character.)

Bit 1: RXRDY. This bit is high when the USART is ready to receive a character.

Bit 2: TXE (Transmitter Empty). This bit is high when the USART is empty (has sent all the characters it has been given to transmit). The USART can accept a character, start sending it, and, while sending, signal TXREADY and accept another character. TXE will not be sent, however, until all characters given the USART have been transmitted.

Bit 3: PE (Parity Error). The PE bit is set when the USART detects a parity error in a character it has received.

Bit 4: OE (Overrun Error).

Bit 5: FE (Framing Error). The FE bit is set when the USART fails to detect a stop bit in a character it has received.

Bit 6: SYNDET. This signal is not used. (It is reserved for use in synchronous data communications.)

Bit 7: Tells whether a high or low is present at the USART's DSR pin (USART pin 22). This pin need not be connected to the RS-232 DSR line, and in the 4027A it is connected to the RING line instead. Thus, by reading bit 7 of the USART Status Word, the Processor can ascertain whether the modem is presenting a RING signal to the terminal.

- **USART Programming.** By writing into the USART Programming Word (the "write" half of the USART Command Register), the Processor can pass commands to the USART.

  After the USART is reset, the first word written here is the **mode instruction word**. This word determines the USART's operating mode.

  The 4027A operates only in asynchronous mode. In that mode, all succeeding words written at this address are treated by the USART as **command instruction words**, as described earlier under "Host Port Firmware."

Baud Rate Word. The Baud Rate Word (Schematic 6-1) is an eight-bit latch. It is loaded from the data bus when clocked by HBRW (Host Baud Rate Write).

Baud Rate Generator. The Baud Rate Generator (Figure 4-18, Schematic 6-1) provides two clocks, RXC and TXC, for the receive and transmit sections of the USART. These are "16X clocks"; their frequencies are 16 times the receive and transmit baud rates. The data in the Baud Rate Word selects which of several standard frequencies are used for the RXC and TXC clocks. (Baud Rate Word bits 0-3 select the TXC frequency; bits 4-7 select the RXC frequency.)

The heart of the Baud Rate Generator is an integrated circuit (Fairchild F4702 or equivalent). This IC includes: a 2.4576 MHz oscillator, frequency dividers to provide 16X clocks for the standard baud rates, and a data selector which selects one of these clocks for output.

Used by itself, the F4702 would provide one of the two USART clocks. Circuitry external to the F4702, however, makes it do double duty, providing two different clock frequencies. This circuitry takes advantage of the fact that the F4702 has internal clock signals for all the standard baud rates, even though only the selected clock is presented to the F4702's Z output.

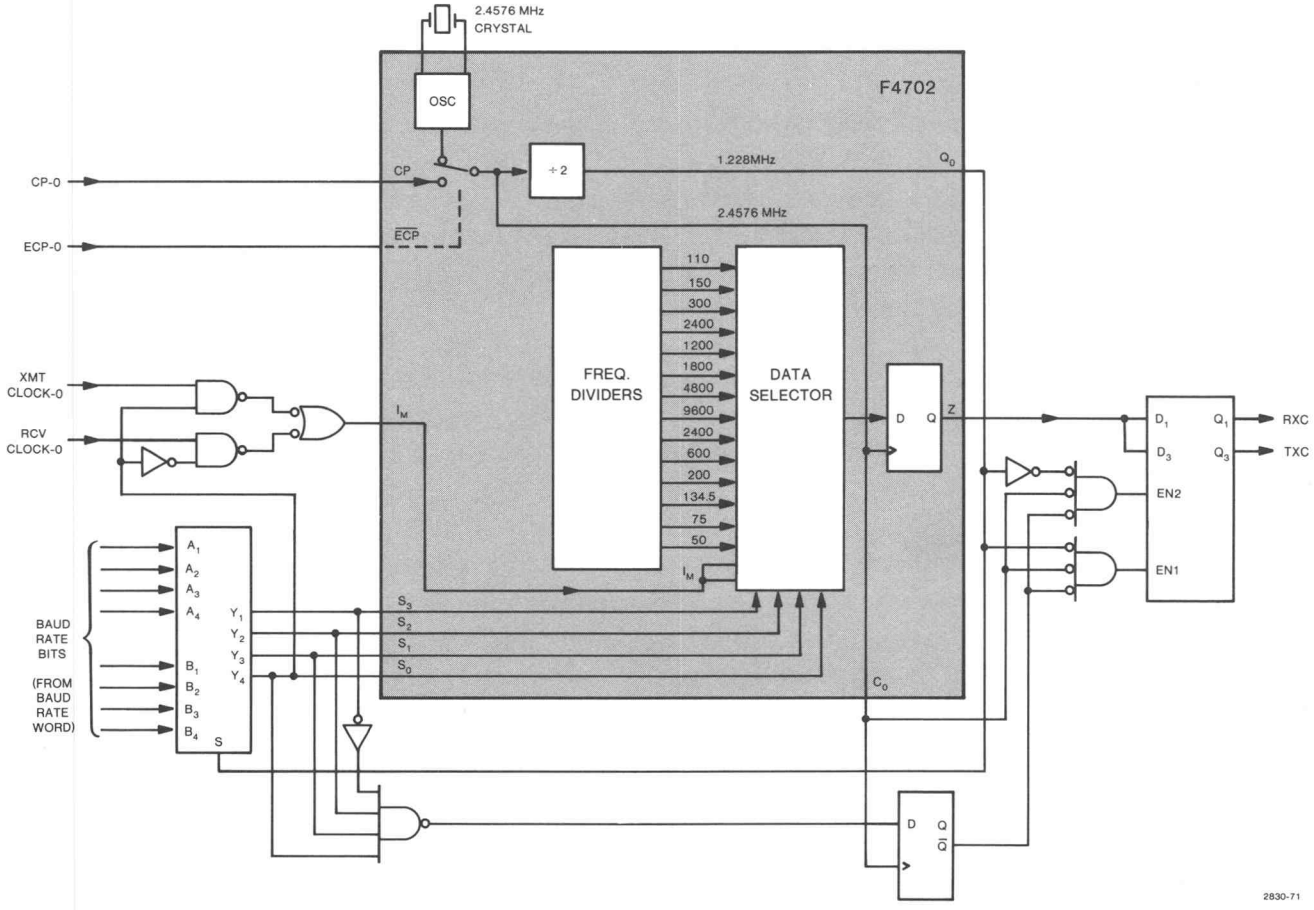


Figure 4-18. Baud Rate Generator.

It works this way:

The F4702 Q0 1.2288 MHz output rapidly switches the F4702's internal data selector between its two internal clocks. The C0 2.4576 MHz output latches the state of the selected clock into one of two flip-flops in a 74LS75. The state of one internal clock is sampled and held on the 74LS75's Q1 output, which drives the RXC line. The other of the two internal clocks is sampled and held on the Q3 output, which drives the TXC line.

As just mentioned, the F4702 Q0 signal determines which of the two internal clocks is being sampled. To do this, the Q0 output steers two devices.

- First, the F4702 Q0 output steers a 74LS73 data selector. When Q0 is low, the data selector places the "receive" bits of the baud rate word on the F4702's speed select inputs S0-S3. When Q0 is high, the data selector places the "transmit" baud rate bits on pins S0-S3.

  When Q0 is low, the internal data selector samples the "receive" baud rate clock; the next oscillator C0 pulse places the state of the "receive" clock on the Z output. When Q0 is high, the internal data selector samples another baud rate clock: the "transmit" baud rate clock. The next C0 pulse places the state of the "transmit" clock on the Z output.

  Thus, the F4702's Z output is multiplexed between the desired two clock signals.

- Secondly, the Q0 output enables one or the other of the two flip-flops which sample the F4702 Z output. When Q0 is low, it enables the Q1 flip-flop, which samples and holds the receive clock waveform. When Q0 is high, it enables the Q3 flip-flop, which samples and holds the transmit clock waveform.

  Thus, the Q0 output is used to control the demultiplexing of the F4702 Z output signal. This demultiplexing separates the two clock signals being alternately presented on the Z output pin. The "receive" clock is gated to the RXC line, and the "transmit" clock is gated to the TXC line.

Figure 4-19 summarizes this process. The Q0 line steers a multiplexer (most of whose parts are within the F4702 integrated circuit). The multiplexer output is the F4702 Z output, which is then demultiplexed to provide the RXC and TXC clock signals. The demultiplexer is steered by the same signal (that is, Q0) as the multiplexer.

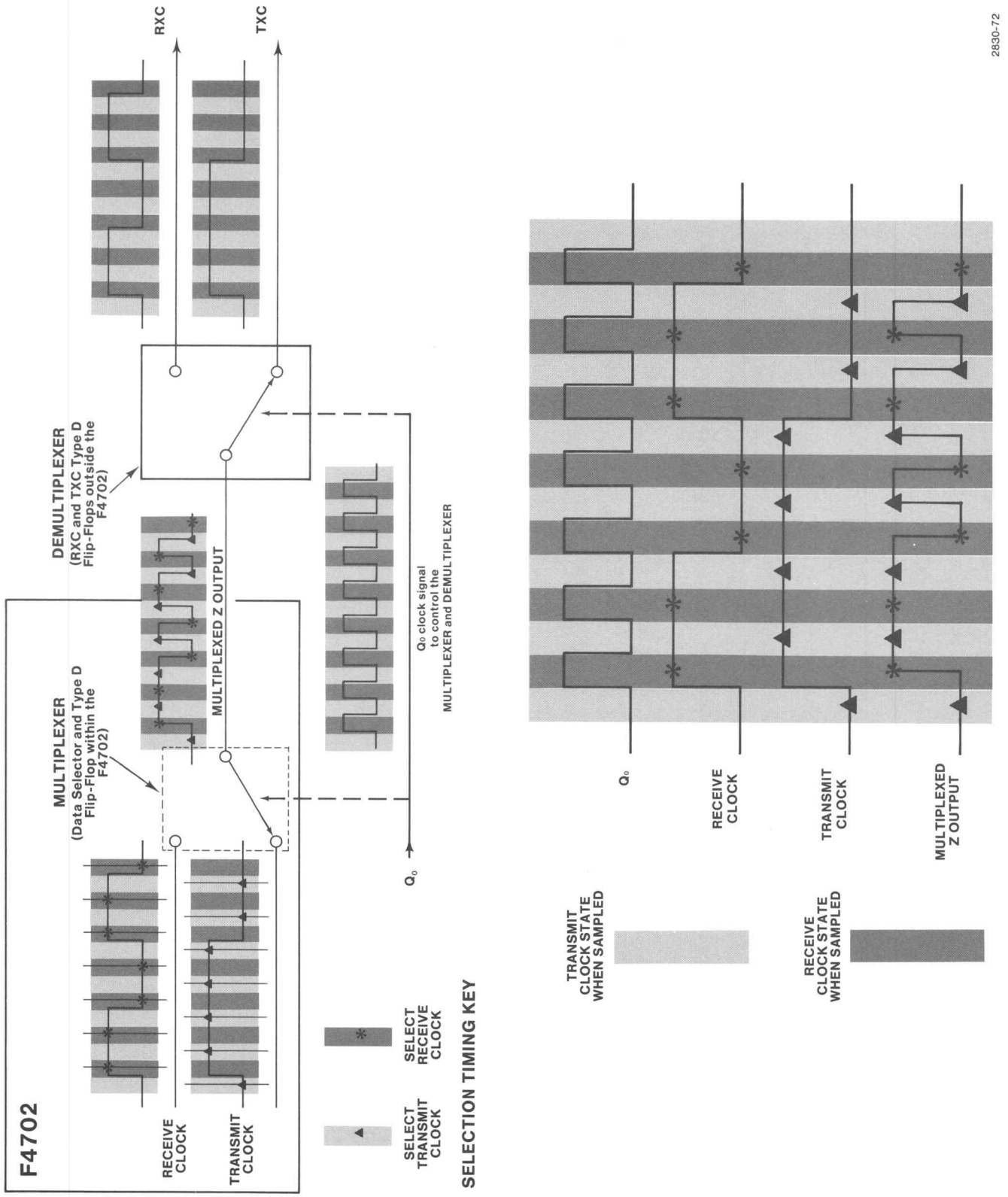


Figure 4-19. Multiplexing and Demultiplexing the Baud Rate Clocks.

The F4702 S0-S3 selector inputs have sixteen possible settings. Three of these have special meanings, as follows:

- Settings binary 0000 and 0001 pass the signal on the F4702 IM pin on the Z output pin. Logic external to the F4702 monitors the S0 input line. When S0 is low, the external RCV CLOCK is applied to the IM input; when SO is high, the external XMT CLOCK is applied.

- Two settings of S0-S3 cause the F4702 to provide a 2400 baud clock. One of these (binary 0111) is not used for this purpose by the firmware. Instead, this setting (0111) is detected by a NAND gate. The NAND gate's output is delayed slightly by a type D flip-flop to match the delay in the F4702 output caused by a similar flip-flop within the F4702. The delayed NAND gate output is used to disable the 74LS75 stopping the baud rate clock.

The result is that the Baud Rate Generator provides 16X clocks for the USART according to the bits stored in the Baud Rate Word, as follows:

bits	7	6	5	4	(transmit)
bits	3	2	1	0	(receive)
	0	0	0	0	RCV CLOCK external clock
	0	0	0	1	XMT CLOCK external clock
	0	0	1	0	50 baud
	0	0	1	1	75 baud
	0	1	0	0	134.5 baud
	0	1	0	1	200 baud
	0	1	1	0	600 baud
	0	1	1	1	0 baud (stops the clock)
	1	0	0	0	9600 baud
	1	0	0	1	4800 baud
	1	0	1	0	1800 baud
	1	0	1	1	1200 baud
	1	1	0	0	2400 baud
	1	1	0	1	300 baud
	1	1	1	0	150 baud
	1	1	1	1	110 baud

RS-232 Receivers and Drivers. The RS-232 Receivers and Drivers are inverting buffers which interface between the RS-232 signal levels (typically +12 V for "high" and —12 V for "low") and the TTL signal levels used within the Deluxe Communications Board (0 V to 0.6 V for "low" and +3.5 V to +5.0 V for "high").

RDATA Logic. The RDATA Logic appears in Schematic 6-3. It controls the local echoing of transmitted data, as follows:

- If data is being transmitted (in which case RTS is true) and the Deluxe Communications Board's Host Port is to provide local echo (Line Protocol bit LP1 is set), then the RDATA signal is not passed to the USART RXD input. Instead, the data being transmitted from the USART TXD output is looped back and presented to the RXD input.

- Otherwise, it is the data coming in on the RDATA line which is presented to the USART's RXD input.

Line Status Word (Read). The "read" half of the Line Status Word is a 74LS368 tri-state buffer (Schematic 6-3). When enabled by the HLSR signal, the states of DSR, SDCD, DCD, and CTS are placed on the data bus.

Line Status Word (Write). The "write" half of the Line Status Word is a 74LS175 integrated circuit holding four type D flip-flops. When clocked by the HLSW signal, it reads data lines D4 to D7 and stores them in its flip-flops. The flip-flop outputs drive the following signals in the Host Port:

- Bit 4 drives the RS-232 DTR (Data Terminal Ready) line.

- Bit 5 can be used to drive the SRTS (RS-232 pin 11) line, if strap "C-D" is wired in position C. Normally, however, this strap is wired in position D, and the Half Duplex Logic drives this line in parallel with the SRTS (RS-232 pin 19) line.

- Bit 6 drives the REQSRTS line. (REQSRTS requests the Half Duplex logic to send the SRTS signal to the modem.)

- Bit 7 drives the RESRTS line. (REQRTS requests the Half Duplex Logic to send the RTS signal to the modem.)

Line Protocol Word. The Line Protocol Word is a write-only register with two bits (data bits 0 and 1). It consists of two type D flip-flops (Schematic 6-3) which are loaded with data bits 0 and 1 when clocked by the HLPW (Host Line Protocol Write) signal. The flip-flop outputs drive the LP0 and LP1 lines, which specify which of several alternate line protocols is to be used. Bit LP1 controls the local echo feature, and the two bits together inform the Half Duplex Logic whether Half Duplex, Half Duplex with Supervisor, or Full Duplex mode is to be used. This is summarized in the following list.

LP1	LP0	Data Communication Mode
0	0	Full Duplex, no local echo. Any echo must be provided by the host (remote echo).
0	1	Half Duplex with Supervisor. Any echo is provided by the firmware, not the Host Port circuitry.
1	0	Full Duplex, local echo provided by the Host Port.
1	1	Half Duplex without supervisor, local echo provided by the Host Port.

Host Port Status Word (Read). The "read" half of the Host Port Status Word (Schematic 6-3) uses five tri-state buffers in a 74LS367. When enabled by the HSWR signal, these buffers place bits on the data bus, as follows:

- Bit 1: XMTRDY. Set when the USART transmitter is ready to accept another character.
- Bit 3: RXRDU. Set when the USART has received a character and is ready for the Processor to read that character from its Data Word.
- Bit 5: LSC (Line Status Change). Indicates when the state of one of the RS-232 signal lines has changed. (When the Processor reads from the Host Port Status Word, this bit is automatically cleared.)
- Bit 6: TXE (Transmitter Empty). Indicates when the USART has finished sending all the characters it has been given to transmit.
- Bit 7: HIRQST. This bit is set whenever the Host Port is requesting an interrupt.

Debouncer. The Debouncer prevents transient noise (such as that due to switch contact bounce) from being detected as line status changes and causing Processor interrupts. The Debouncer does this for the RING, DSR, DCD, CTS, and SDCD RS-232 signal lines, and for the SYNDET signal from the USART.

The Debouncer (Schematic 6-3) consists of six type D flip-flops in a 74C174 integrated circuit. The flip-flops act as one-stage shift registers, so that their outputs lag one LCLK pulse behind their inputs. In addition, the slow response time of the 74C174's CMOS circuitry removes any short-duration noise. Consequently, the Q outputs of the flip-flops are noise-free, but lag slightly behind the D inputs. The LSC Detector takes advantage of this slight delay.

LSC Detector. The LSC (Line Status Change) Detector is shown in Schematic 6-3. It consists of several NOT-XOR gates and a flip-flop.

Each NOT-XOR gate is connected to a Debouncer input and the corresponding Debouncer output. When the status of that particular line changes, there will be a brief period when, due to the propagation delay through the Debouncer, the input and output have different states. When that occurs, the NOT-XOR gate's open-collector output goes low.

The NOT-XOR gates are connected in a "wired-OR" configuration to drive the preset input of the flip-flop. Thus, the flip-flop is set whenever a line status change occurs on the RING, DSR, SYNDET, DCD, CTS, or SDCD signal lines. The flip-flop will be cleared when the Processor reads from the Host Port Status Word (by the HSWR signal) or on power-up or RESET (by the CLEAR signal).

Host Port Interrupt Requestor. The Host Port Interrupt Requestor (Schematic 6-3) consists of three open-collector NAND gates connected in a wired-OR configuration. These gates send the HIRQST (Host Port Interrupt Request) signal whenever any of the following occur:

- The XMTRDY interrupt is enabled (Host Port Status Word bit 0 is set) and the XMTRDY signal occurs.
- The RXRDY interrupt is enabled (Host Port Status Word bit 2 is set) and the USART sends the RXRDY signal.
- The LSC interrupt is enabled (Host Port Status Word bit 4 is set) and the LSC Detector sends the LSC signal.

Line Protocol Logic. (Schematic 6-2) The Line Protocol Logic controls the "handshaking" (exchange of control signals) between the terminal and its modem. Different handshaking rules apply in each of the three communications modes: full duplex, half duplex normal (without supervisory channel), and half duplex with supervisor.

- **Full Duplex Mode.** This mode is selected when the Line Protocol bit LP0 is zero. The RS-232 RTS, SRTS, and DTR signals are under direct firmware control. (The bits in the write half of the Line Status Word are passed directly to their respective RS-232 signal lines.) The USART's TXRDY signal is passed directly to the XMTRDY line, so that the Processor sees an OUTRDY state whenever the USART is ready to accept another character.

- **Half Duplex, Local Echo.** (LP1 and LP0 are both ones.) RTS, SRTS, and DTR are under direct firmware control. The Processor sees an OUTRDY condition only when the USART is sending its TXRDY signal and modem is sending CTS. Whenever RTS is true, transmitted data is echoed back to the receiver.

- **Half Duplex With Supervisor, No Echo.** The DTR signal remains under direct firmware control. The RTS and SRTS signals are controlled by the Line Turnaround circuit, described later. The Processor sees an OUTRDY only when all the following conditions are met: (a) the Line Turnaround Logic is in its "transmit" state (RTS true, SRTS false), (b) the USART is ready (TXRDY true), and (c) the modem is signaling "clear to send" (CTS true).

  When a BREAK is received (and the Line Turnaround Logic is in the "receive" state), SRTS will go false to notify the host computer. (As SRTS goes false, the modem stops sending the secondary carrier. This causes the computer's modem to stop sending the SDCD signal to the computer.)

A 74LS153 dual data selector controls the OUTRDY and SRTS signals. Line Protocol bits LP0 and LP1 steer the data selector as follows:

- LP1=0, LP0=0; Full duplex, No Echo. The data selector's C0 inputs are selected. OUTRDY is driven by the USART's TXRDY signal, and SRTS is driven by the REQSRTS bit from the Line Status Word.

- LP1=1, LP0=0 or =1; Full Duplex, Local Echo or Half Duplex, Local Echo. The data selector's C2 or C3 inputs are selected. OUTRDY is sent only when the USART is empty (TXRDY true) and CTS is true. As before, SRTS is driven by the REQRTS bit from the Line Status Word.

- LP1=0, LP0=1; Half Duplex Supervisor. The data selector's C1 inputs are selected. OUTRDY and SRTS are driven by the Line Turnaround Logic.

The RTS signal is driven by a logic tree which uses a 74LS51 AND-OR-INVERT gate, a negative-logic OR gate (inverting input, inverting output OR gate), and an inverter.

- Whenever LP0=0 or LP1=1, half duplex supervisor mode is not being used. In this case, the RTS signal is driven by the REQRTS signal from the write half of the Line Status Word.

- When LP0=1 and LP1=0, the terminal is in half duplex supervisor mode. In this case, RTS is governed by an output of the Line Turnaround Logic.

Line Turnaround Logic. (Schematic 6-2) The Line Turnaround Logic functions when the terminal is in "half duplex supervisor" mode. In this mode, data transfer occurs in only one direction at a time; while the 4027A is sending to the computer, the computer cannot send back to the 4027A and vice versa. The Line Turnaround Logic controls the process of changing the direction of data transfer.

The Line Turnaround Logic (Schematic 6-2) has four different states or conditions during the data transmission cycle. These states are represented by the four possible states of flip-flops Q0 and Q1 as follows:

State Name	Q1	Q0	Description
A	0	1	"Receive" state.
B	1	1	Changing from "receive" to "transmit."
C	1	0	"Transmit" state.
D	0	0	Changing from "transmit" to "receive."

The Line Turnaround Logic is implemented as a "state machine." Flip-flops Q0 and Q1 designate the states. TTL logic gates change the state machine from one state to another by controlling flip-flop inputs. Figure 4-20 is a state diagram summarizing the design of this state machine. The following text explains the state diagram in more detail:

- **State A ("Receive" State).** The Line Turnaround Logic remains in "receive" state so long as the modem continues to send the DCD (Data Carrier Detect) signal. (DCD true means the computer's modem is using the data communications channel.)

  When the computer stops sending data to the terminal, it turns off its data carrier. The terminal's modem responds by turning off the DCD signal. When the Line Turnaround Logic sees DCD drop, it exits State A and enters State B.

- **State B (Changing From Receive to Transmit).** In State B, the Line Turnaround Logic sends the RTS signal, requesting the modem to place a carrier tone on the telephone line. The Line Turnaround Logic then waits about 1.5 seconds in State B for a secondary "supervisory" carrier on the telephone line. The secondary carrier indicates that the computer is ready to receive data from the terminal.

  After the 1.5 second delay, the Line Turnaround Logic exits State B and enters State C.

- **State C ("Transmit" State).** In State C, the Line Turnaround Logic continues to send the RTS signal. It is during this state that the terminal transmits data to the computer. The Line Turnaround Logic remains in "transmit" state so long as the SDCD signal is present.

  Should the computer need to use the data communications channel, it will disable the secondary carrier tone. Consequently, the 4027A's modem will stop sending SDCD. The Line Turnaround Logic waits until the USART has emptied its transmit buffers and sent the TXE signal. As soon as SDCD is false and TXE is true, the Line Turnaround Logic exits State C and enters State D.

- **State D (Changing From Transmit to Receive).** In State D, the Line Turnaround Logic stops sending RTS and waits about 300 ms to give the computer's modem a chance to respond by placing a carrier on the telephone line. After the 300 ms delay, the Line Turnaround Logic exits State D and enters State A.

On power-up, the CLEAR signal resets both flip-flops, putting the state machine in State D. The J and K flip-flop inputs are: J1=0, K1=1, J0=DELAY1, K0=0. The 200 ms timer, triggered by flip-flop Q1's inverting output, starts to time a 200 ms pulse. During the 200 ms period, DELAY1 is false, keeping J0=0. With J0=K0=0, Flip-Flop Zero does not change state, and the state machine stays in State D.

When the one-shot times out, however, DELAY1 goes true, and J0 goes high. The next LCLK pulse sets the flip-flop, and the state machine enters State A.

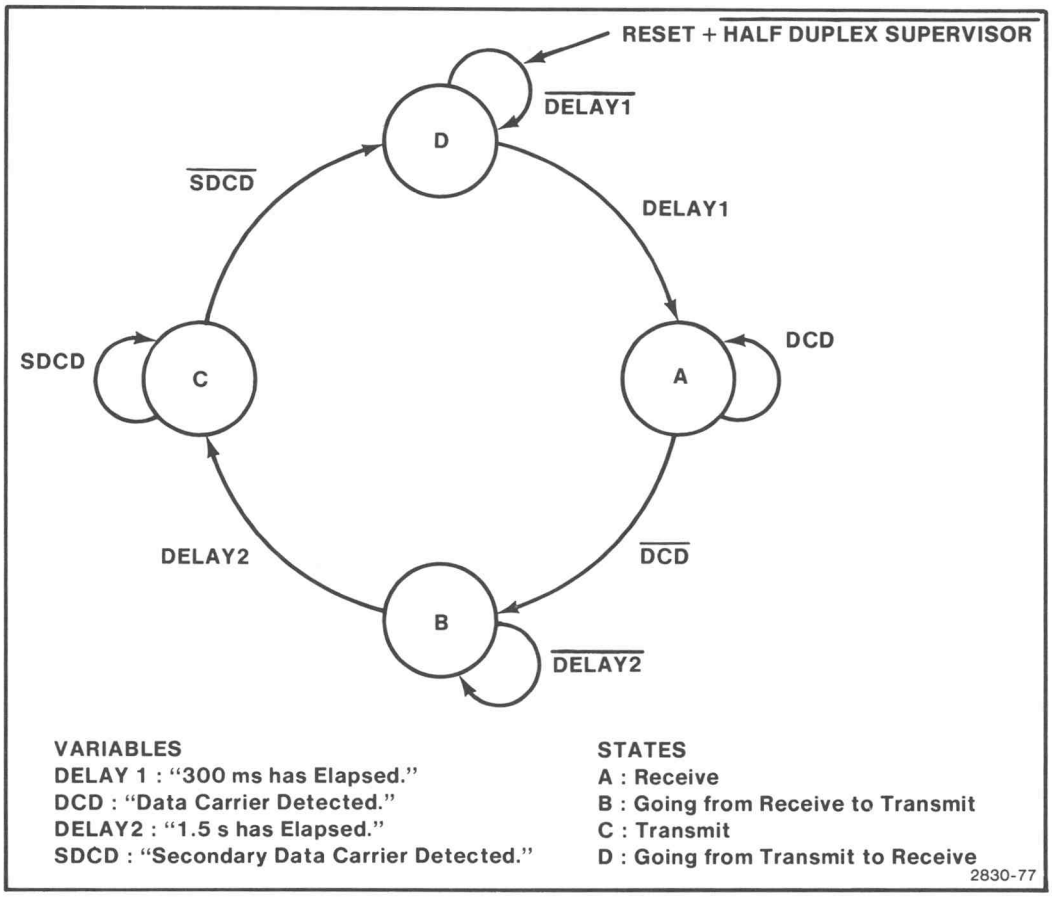


Figure 4-20. Line Turnaround State Diagram.

In State A, the "receive" state, we have J0=1, K0=0, J1 = not DCD, K1=0. That is, flip-flop Q0 will not change state since its J and K inputs are both low, and flip-flop Q1 will not change state as long as DCD is true (and "not DCD" at its J input is low). Thus, as long as the computer's modem sends the data carrier (and the terminal's modem asserts DCD), the state machine remains in the "receive" state.

When DCD goes false, Q1's J input goes high. The next LCLK pulse sets the flip-flop, and the state machine enters State B.

Entering State B triggers the one-second timer. This one-shot sends DELAY2 false until it times out, whereupon it sends DELAY2 true. In State B, the flip-flop J and K inputs are: J1="not DCD," K1=0, J0=0, K0=DELAY2. Since flip-flop Q1 is already set in State B, what happens to its J input is irrelevant; the flip-flop stays set whether DCD is true or false. The K input of Flip-Flop Zero, however, does matter. So long as DELAY2 is false, the Q0 flip-flop stays cleared. When the one-shot times out, K0=DELAY2 goes true, and the flip-flop is set. This moves the state machine into State C.

In State C (the "transmit" state), J1=0, K1="not SDCD and TXE", J0=0, and K0=1. Since J0=0 and K0=1, flip-flop Q0 stays cleared. Flip-flop Q1 stays set only as long as its K1 input stays low. That is, the state machine stays in State C only so long as either the USART transmit buffer is not empty (TXE false) or a secondary "supervisory" carrier is being received from the computer's modem (SDCD true).

Should the host computer need to use the data communication line, it will send a "break" by turning off the secondary carrier. SDCD goes false (and "not SDCD" goes true). As soon as the 4027A USART is done transmitting (signalled by TXE), J1 goes true. The next LCLK pulse will then clear the flip-flop, putting the state machine in State A.

As noted before, the state machine stays in State A for 300 ms, and then enters State B, the "receive" state.

Keyboard Port

The keyboard sends and receives data in a serial format. It repeatedly sends (on the KDATA line) a 128-bit "character" whose individual bits tell whether keyswitches are open or closed. The Keyboard Port monitors the incoming stream of KDATA bits and interrupts the Processor whenever a key is pressed or released. The Processor then learns which key has been pressed or released by reading from the Keyboard Data Word.

CLOCK and SYNC signals from the Keyboard Port keep the stream of KDATA bits synchronized with the Character Counter which decodes them.

Likewise, the Keyboard Port uses a serial format to send "lights and bell" data to the keyboard on the LDATA line. It provides the SCLK (Shift Clock) signal for clocking the LDATA bits into the keyboard's shift register and latches.

Keyboard Port Firmware

The following description explains how the Keyboard Port circuitry is used. Two topics are described: how the Processor reads characters typed on the keyboard, and how it controls the keyboard lights and bell.

As keys are pressed or released, the following occurs:

1. One of the bits changes in the 128-bit "character" being sent from the keyboard on the KDATA line. The bit's new state shows the new position of the key.

2. The Keyboard Port monitors the stream of KDATA bits, and notes the change in the bit representing that key. It stores an 8-bit word in the "read" half of the Keyboard Data Word (address '0832'). Seven of the word's bits name the key which moved. The eighth bit tells whether the key was pressed (bit = 1) or released (bit = 0).

3. Simultaneously, the Keyboard Port requests a Processor interrupt.

4. The Processor interrupts its current task and reads from address '0832'. It consults a table (stored elsewhere in memory) to ascertain the key's definition. The definition may be an ASCII character (a binary number between 0 and 127) or a function key code (a binary number between 128 and 255). If the key has been programmed, its definition may be a string of several ASCII characters or function key codes.

5. The Processor stores the key's definition in its input queue (another part of memory).

6. The Processor returns from the interrupt handling routine and resumes its previous task.

Later the Processor examines its input queue and does one or more of the following:

- Places characters from the input queue in the display list, so that they may be displayed on the screen.
- Places characters from the input queue in the transmit buffer, so that they may be sent to the host computer.
- Interprets strings of characters as commands and executes the commands.

The Processor controls the keyboard lights and the terminal's bell (which is in the keyboard) by writing into the Keyboard Data Word, address '0832'. In this word, bit 0 is set if the bell is to be rung, and bits 4 through 7 each control one of the four function key lights. Bits 1 through 3 are unused.

Thus, to ring the bell the Processor first writes into '0832', setting bit 0 = 1. To turn off the bell, it writes into '0832', setting bit 0 = 0.

Keyboard Port Circuitry

Figure 6-8 is the block diagram for the Keyboard Port. Five circuit blocks process data coming from the keyboard. These are the Clock Counter, Keyswitch Counter, Sync Generator, Key Status Change Detector, and Keyboard Data Word (Read part). The remaining blocks control the keyboard lights and bell. These are the Keyboard Data Word (Write), SCLK Generator, and Shift Register. Each block is discussed in turn.

Clock Counter. (Schematic 6-4) The Clock Counter divides the 2.048 MHz clock (LCLK) by 16 to provide the 128 kHz CLOCK signal for the Keyboard and the Keyboard Port. The CLOCK is stopped (counter cleared) whenever a change in positions of the keyswitches is detected (KEYRDY condition). (When the Processor reads from Keyboard Data Word, the KEYRDY condition is cleared and the CLOCK starts again.)

The counter is also cleared on power-up or when the RESET switch is pressed, and the Processor may clear it by writing into the Keyboard Reset Word.

Keyswitch Counter. (Schematic 6-4) The Keyswitch Counter is stepped by the CLOCK pulses together with a similar counter within the keyboard's Key Scanner. The least significant 7 bits of the count (outputs 1QA to 2QC) specify which key is currently being scanned. That is, these 7 bits tell which key's position is currently being reported on the KDATA line.

The counter's most significant bit (output 2QD) is used to trigger the Sync Generator.

Sync Generator. (Schematic 6-4) The Sync Generator consists of two flip-flops which provide a SYNC signal for the Keyboard and a KBDSYNCD signal for the Keyboard Port.

The first flip-flop is set when the Keyswitch Counter's 2QD output goes high. At this time, the Keyswitch Counter's least significant bits are all zero, and the similar counter in the Keyboard's Key Scanner is reset to zero. The flip-flop output is the SYNC pulse for the keyboard's Key Scanner.

The second flip-flop drives the KBDSYNCD (Keyboard Synchronized) line. On power-up, or after the Processor resets the Keyboard Port, this flip-flop is cleared, sending KBDSYNCD false. The first SYNC pulse to occur sets the flip-flop, sending KBDSYNCD true. Until KBDSYNCD goes true, the Host Port will not generate KEYRDY interrupts.

Key Status Change Detector. (Figure 4-21, Schematic 6-4) The Key Status Change Detector looks for changes in the keyswitch positions reported on the KDATA line. On detecting a valid change in keyswitch position, it sets a flip-flop and sends the KEYRDY signal. This stops the CLOCK (clears the Clock Counter) so that the Character Counter holds its current count, which can be read from the Keyboard Data Word. If KEYRDY interrupts are enabled, the Processor reads from that word, thus learning which key's position has changed.

To examine changes in any key's position, the Key Status Change Detector must compare each KDATA bit with the KDATA bit 128 CLOCK pulses earlier. (The KDATA bit 128 CLOCK pulses earlier represents the same key's position the previous time it was scanned.) In fact, to eliminate the effects of keyswitch bounce, the circuit compares the key's position with its position on several previous scans of the keyboard. To do this, it uses a type 5055 integrated circuit, which contains four 128-stage shift registers.

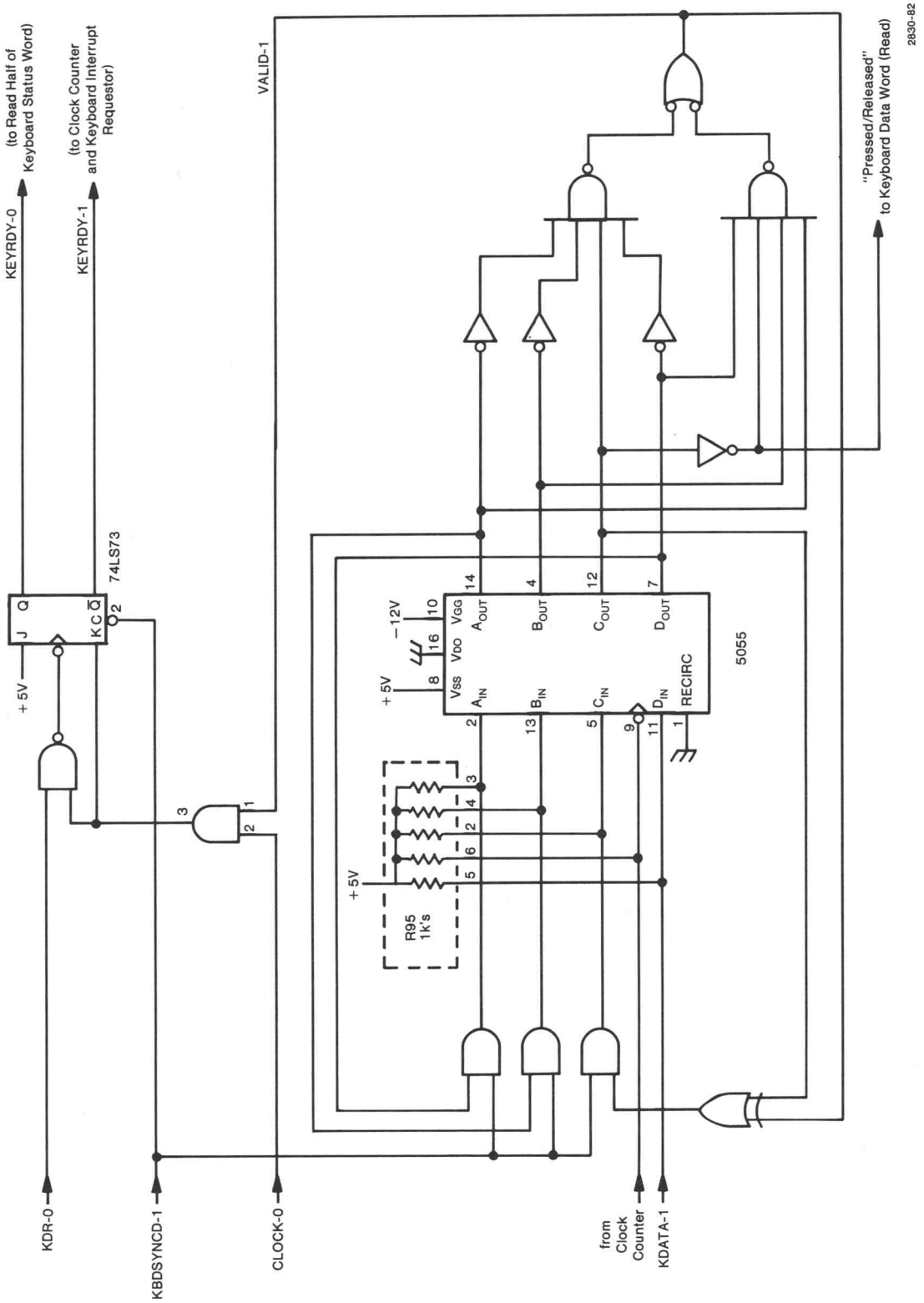


Figure 4-21. Key Status Change Detector.

The KDATA signal goes to the D input of the 5055. After 128 CLOCK pulses, it emerges from the D output. Thus, the D output shows the "past history" of the key currently being scanned: its position on the previous scan of the keyboard. Provided KBDSYNCD is true, the D output is delayed three more 128-clock pulse periods in the A, B, and C sections of the 5055. Thus the 5055's D, A, B, and C outputs show the state of a key (the key currently being scanned) during the previous four times it was scanned.

Two four-input NAND gates monitor the shift register outputs. They detect when a key has held the same state for three consecutive scan periods (the D, A, and B outputs agree) but its state was different the scan period before that (the C output differs from the D, A, and B outputs). One NAND gate detects low-to-high transitions (keys being pressed); the other gate detects high-to-low transitions (keys beng released). By requiring that the key's state be the same on three consecutive scans of the keyboard, the gates reject false readings due to switch contact bounce.

The NAND gate outputs are fed to an OR gate, which provides a VALID signal when a valid change of keyswitch position has occurred. VALID sets the KEYRDY flip-flop. The KEYRDY signal, in turn, stops the Clock Counter, so that the keyswitch code is held static in the Keyswitch Counter. KEYRDY also interrupts the Processor, so that it may read the Keyboard Data Word.

Keyboard Data Word (Read). (Schematic 6-4) The "read" half of the Keyboard Data Word is a 74LS241 integrated circuit holding eight tri-state buffers. When the Processor (in response to a KEYRDY interrupt) reads from address '0832', the KDR signal enables these buffers, placing the keyboard data word on the data bus. The least significant seven bits of this word (from the Keyswitch Counter) show which key was pressed or released. The most significant bit (from the Key Status Change Detector) shows whether the key was pressed (bit 7 = 1) or released (bit 7 = 0).

Keyboard Data Word (Write). (Schematic 6-4) The "write" half of the Keyboard Data Word is a type 74LS273 latch. When the Processor writes into address '0832', the KDW signal clocks this latch, storing the eight bits being presented on the data bus lines D0-D7. The latch outputs Q1 to Q8 drive the Shift Register's parallel inputs.

Shift Register. (Schematic 6-4) The Shift Register converts the parallel data from the "write" half of the keyboard Data Word into the serial LDATA signal for the keyboard. It is loaded by the LOAD signal from the SCLK Generator, and its data is shifted out by the SCLK (Shift Clock) signal.

SCLK Generator. (Figure 4-22, Schematic 6-4) The SCLK Generator generates the SCLK and LOAD waveforms. LOAD is used only to load the Shift Register in the Keyboard Port. SCLK, on the other hand, serves both the Keyboard Port and the keyboard itself. SCLK has three functions:

- It clocks LDATA bits out of the Keyboard Port's Shift Register.
- It clocks those same bits into a shift register in the Keyboard.
- It is used to load a latch in the Keyboard, in which the LDATA bits are stored.

Each group of eight SCLK pulses shifts the LDATA bits out of the Shift Register, sending them to the keyboard. The inverted SCLK waveform then clocks these same bits into another shift register within the Keyboard.

After the eight LDATA bits have been shifted out, a LOAD pulse occurs, and the SCLK pulses are turned off. LOAD reloads the Keyboard Port's Shift Register. The absence of LCLK pulses during this time is detected by the keyboard, which transfers the word just clocked into its shift register into a latch. (This is described in more detail in the keyboard circuit description.)

The SCLK generator is comprised of a 74393 dual divide-by-16 counter and an AND gate. The first half of the counter divides the 2.048 MHz LCLK signal by 16, giving a 128 kHz square wave at its 1QD output. This is again divided by 16 in the second half of the counter, to provide the LOAD waveform at the 2QD output. The AND of these two waveforms is the SCLK waveform.

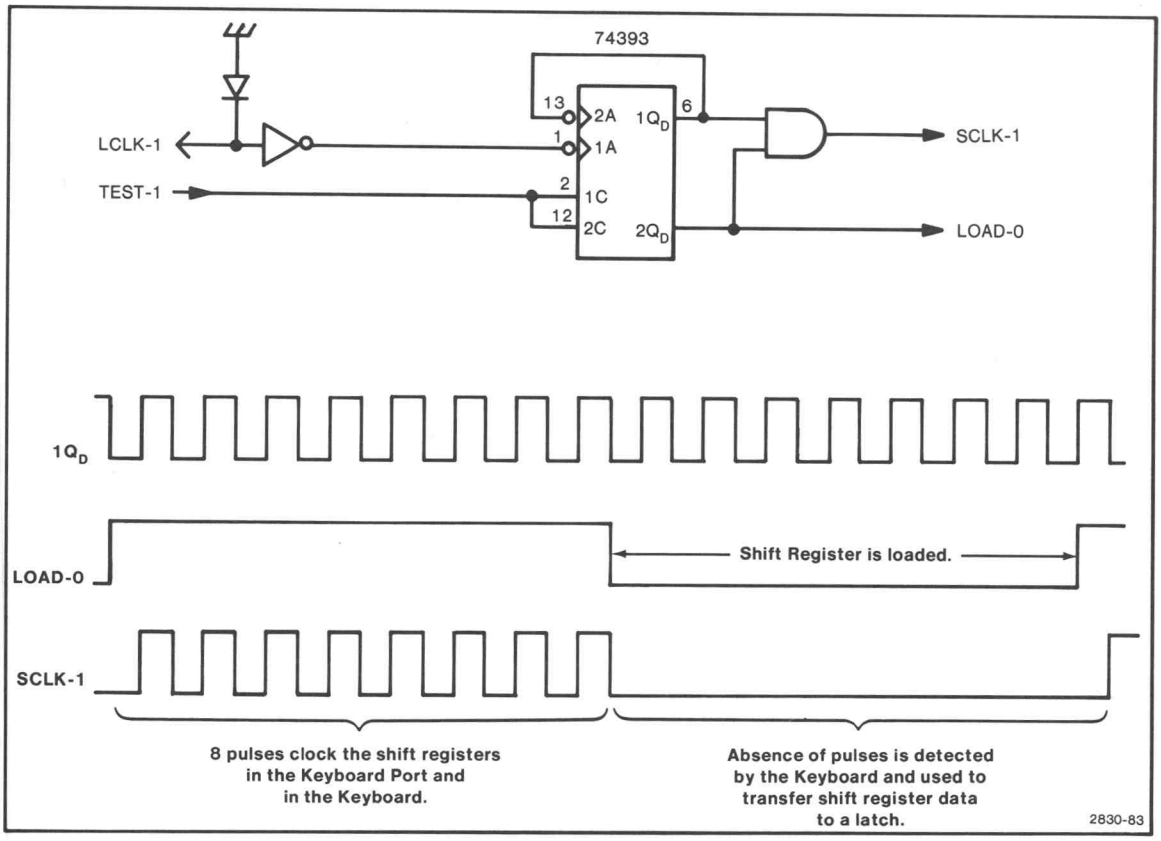


Figure 4-22. SCLK Generator and Waveforms.

KEYBOARD

The keyboard (Schematic 8-1) is connected to the Keyboard Port of the Deluxe Communications Board by an 8-conductor cable. Its circuitry has two major functional blocks: the Key Scanner and the Lights and Bell circuitry.

Key Scanner

The Key Scanner tells the keyboard port which keys are pressed. A CLOCK signal from the Deluxe Communications Board drives a 7-bit binary ripple counter. The four low-order bits of the counter are used by a 74L154 multiplexer to scan the 16 columns of a keyswitch matrix. The three high-order bits are used by a 74LS151 to scan the eight rows of the matrix. Thus, each keyswitch is interrogated once during each 128-count cycle of the counter.

For example, the W switch is interrogated on the count of 35 (binary 010 0011), when the fourth column (74L154 output D3) and third row (74LS151 input D2) are both selected. If the switch is closed, the low which the 74L154 presents on the column line drives the D3 row input of the 74LS151 and Pin 6 of the 74LS151 goes high.This drives the KDATA (key data) line low. The fact that it is the 35th count when KDATA goes low tells the keyboard port that it is the W switch which is pressed.

Once each 128 CLOCK pulses, Keyboard Port clears the counter with a SYNC pulse. This ensures that the counter is always in step with a similar counter in the Keyboard Port.

Lights and Bell

The Lights and Bell circuitry receives a multiplexed LDATA (light data) signal from the communications interface, together with SCLK (shift clock). LDATA is demultiplexed and used to drive the lights in the four lighted function keys and sound the bell.

SCLK clocks the LDATA signal into U5, a 74LS164 shift register. SCLK's waveform is shown in Figure 4-23. It consists of eight clock pulses followed by a period, equal in duration to those eight pulses, during which SCLK is turned off. During the eight clock pulses, the open-collector output of inverter U96G is repeatedly driven low, and there is not enough time between pulses for capacitor C4 to charge. However, after the string of eight pulses, SCLK remains off for a longer period, so that C4 can charge and turn on U10E, clocking the type D latches in U6.

Each group of eight SCLK pulses then clocks an eight-bit serial LDATA word into the shift register, and the gap between that group of pulses and the next group clocks the parallel outputs of the shift register into the type D latches in U6. The outputs of these latches drive the four LEDs in the lighted function keys and control the bell.

The "bell" consists of an oscillator (U7) and an output stage. The oscillator runs continuously. The Q6 output of the latch, U6, keys the oscillator output, turning the speaker on and off. When the speaker is turned on, C5 charges; when it is turned off, C5 discharges gradually so that the speaker output decreases gradually, rather than abruptly. This gives a bell-like quality to the tone.

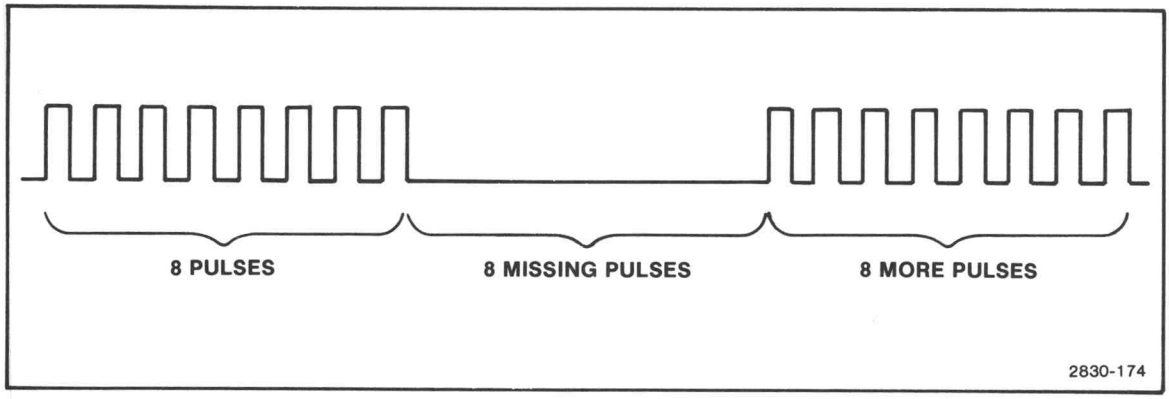


Figure 4-23. SCLK Waveform.

MOTHER BOARD

The Mother Board (Schematic 7-1) carries the Processor's bus, power supply voltages, and various signal lines for the circuit boards that make up the Display Processor portion of the terminal.

Figure 4-24 shows the connector pin identification, and Table 4-8 lists the signal lines on the Mother Board.

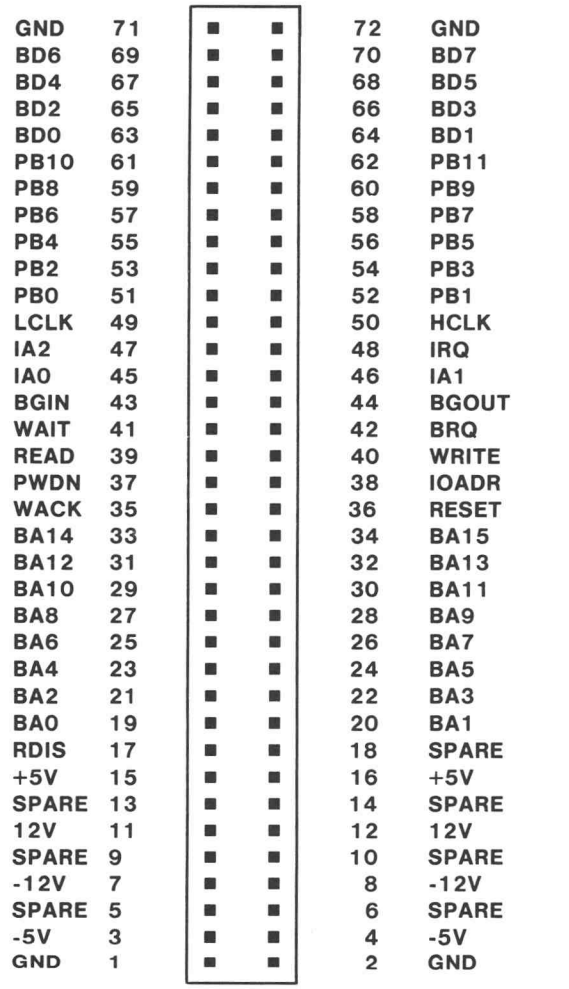


Figure 4-24. Mother Board Pin Identification.

Table 4-8

MOTHER BOARD SIGNAL LINES

Name	Pin No.	Description
P/S Lines	1-16	Pins 1 and 2, at the rear end of the bus, carry grounds.
RDIS	17	The RDIS (ROM Disable) line is normally high. Putting this line low disables the ROM at X'6000'.
BA0-BA15	19-34	The 16 lines BA0 (Bus Address line 0) to BA15 carry the 16 address bits used by the microprocessor to specify any word in its memory. These lines are normally driven by the Processor Board.
WACK	35	The Processor sends a WACK (Wait Acknowledge) signal to the device sending the WAIT signal. The WACK signal indicates that the WAIT has been received.
RESET	36	The RESET line is driven low by the Processor Board during power-up or when the MASTER RESET button is pressed. This signal resets all boards on the bus to a known initial state.
PWDN	37	The PWDN (Power Down) signal, from the Power Supply indicates that primary power has been lost.
IOADR	38	The IOADR (Input/Output Address) line, when in the high state, indicates that the address on the bus is in the range 0800 to 0BFF. This is the section of memory reserved for device registers.
READ	39	The READ signal indicates the start of a read operation. This signal is driven by the Processor Board.
WRITE	40	Similar to the READ signal, but used for write operations.
WAIT	41	The WAIT line, when low, tells the Processor that the addressed device cannot complete the required operation at the moment. As soon as the device can complete the operation, it should release this line.

Table 4-8 (cont)

MOTHER BOARD SIGNAL LINES

Name	Pin No.	Description
BRQ	42	
BGIN, BGOUT	43-44	
IA0-IA2	45-47	The IA0 (Interrupt Address 0), IA1, and IA2 lines are part of the interrupt servicing hardware. They indicate, by a three-bit binary number, the level of interrupt that is being polled. IA2 is the most significant bit.
IRQ	48	A device pulls the IRQ (Interrupt Request) line low when it needs to generate an interrupt and when the IA2-IA0 lines are at its selected interrupt level.
LCLK	49	The LCLK (Low Frequency Clock) line carries the 2.048 MHz clock.
HCLK	50	HCLK (High Frequency Clock) is an 18.432 MHz clock signal.
PB0-PB11	51-62	The twelve "paired bus" lines do not run the entire length of the bus; instead, they connect pairs (in one case, four) of adjacent boards. Care should be taken that the boards involved are in their proper slots.
BD0-BD7	63-70	The lines BD0 (Bus Data) to BD7 make up the 8-bit data bus. These lines are driven by any device placing data on the bus. For example, they are driven by the Processor Board during Processor write operations.
GND	71-72	Like Pins 1 and 2 at the rear of the Mother Board, Pins 71 and 72 at the front of the board carry grounds.

DEFLECTION BOARD

The Deflection board receives horizontal and vertical sync inputs and generates the currents required to drive the horizontal and vertical deflection yoke. It also generates HRAMP (Horizontal Ramp), VRAMP (Vertical Ramp), Dynamic Focus, and High Voltage Inhibit which are used in other areas of the Color Video Display circuitry. For the following discussion, refer to Schematics 9-1 and 9-2 and the Color Video Display Block Diagram, Figure 6-10.

Vertical Ramp

The Vertical Ramp circuit generates a linear voltage ramp which swings between approximately —4 volts and +4 volts (depending on the setting of the VSIZE adjustment). The input stage in the Vertical Ramp circuit is a relaxation oscillator which free-runs at a frequency slightly lower than the VSYNC-1 rate. This oscillator locks on to VSYNC-1. The output of the oscillator triggers a Miller Integrator circuit by turning on Q125. This discharges the integrating capacitor C126. Once discharged, C126 begins charging through Q127, thereby producing VRAMP, a linear, 60 Hz voltage ramp. The setting of V. LIN. TOP (R119) determines the starting point of the ramp.

There are two outputs from the Vertical Ramp circuit. One is produced by a simple voltage follower. This is VRAMP FIXED which is used in the convergence circuits. The other output is taken across the VSIZE control (R125) and goes to the Geometry and Vertical Linearity circuit.

Geometry and Vertical Linearity

The Geometry and Linearity circuit does three things: first, it adds an "S" correction to the vertical ramp to compensate for nonlinearity. Secondly, it adds a horizonal rate, parabola-like waveform to the vertical ramp to correct pincushion distortion at the top and bottom of the raster.

The Geometry and Vertical Linearity circuit also sends a composite geometry correction waveform (GEOMETRY) to the Horizontal Picture Width Regulator and the Dynamic Focus Amplifier. This waveform is used to correct pincushion distortion at the sides of the display and to dynamically maintain electron beam focus at all points on the crt screen.

Vertical Deflection Amplifier

The corrected VRAMP output from the Geometry and Vertical Linearity circuit goes to the Vertical Deflection Amplifier where it is summed with a DC offet voltage from the vertical position control (VPOS R351). The output of the vertical deflection amplifier to the deflection yoke is a current ramp, inverted with respect to the voltage ramp input.

Current feedback is provided by sampling the voltage across a 1-ohm resistor (R178) in series with the ground side of the yoke. This voltage is then summed at the input to the amplifier.

The feedback sample is also sent to the High Voltage Inhibit circuit where it is used to detect the presence or absence of vertical deflection.

Horizontal Phase Shifter

HSYNC-0 from the Display Controller board provides the clock for the horizontal deflection circuits. HSYNC-0 triggers a delay circuit consisting of a 74122 dual one-shot, and its associated passive components. This circuit produces a pulse which is delayed by an amount controlled by the horizontal phase control (HPHASE, R205). The delayed signal is sent to the Phase Comparator and Horizontal Oscillator circuit.

The Phase-Locked Loop

The Horizontal Oscillator and Phase Comparator, the Flyback Base Driver, the Horizontal Sweep circuit, the Flyback Pulse Regenerator, and the Up-Down Ramp circuit constitute a phase-locked loop. The PLL maintains a constant phase relationship between the delayed HSYNC-0 signal and the horizontal sweep, thereby synchronizing the sweep to the display video.

Horizontal Oscillator and Phase Comparator

The heart of the PLL is the Horizontal Processor I.C. (U217) which contains an oscillator and a phase comparator. This device compares the incoming delayed HSYNC-0 to the up-down ramp. The Up-Down Ramp circuit is controlled by the Flyback Pulse Regenerator which is, in turn, triggered by the horizontal sweep flyback voltage. If any phase change occurs, the Horizontal Processor corrects for the phase change by varying the frequency of its internal oscillator. As long as there is an incoming HSYNC-0 signal and a horizontal sweep, the system remains phase-locked. If HSYNC-0 is lost, or the feedback path is broken, the oscillator will free-run.

Flyback Base Driver

The Flyback Base Driver circuit receives a fixed duty cycle switching output from the Horizontal Processor. It transforms this output to the necessary switching levels to drive the base of the flyback transistor, Q1102 (by way of transformer T571).

Horizontal Sweep

The Horizontal Sweep Circuit is of the resonant flyback retrace type. As shown in Figure 4-25, one end of the horizontal yoke is connected to a variable voltage supply of roughly 95V. This is the Horizontal Sweep Supply and Picture Width Regulator. The other end of the yoke (the flyback line) connects to the collector of the flyback transistor, Q1102.

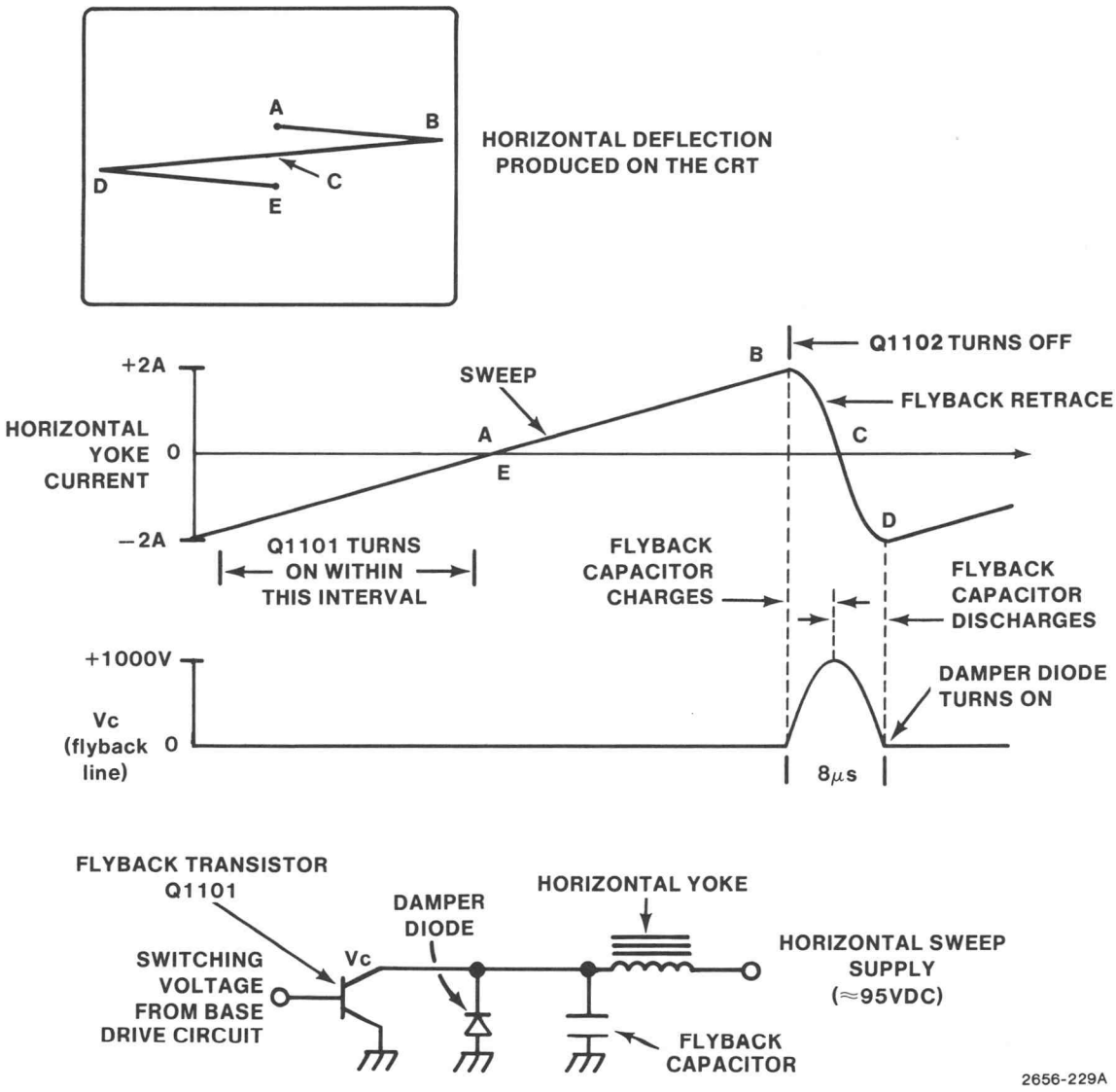


Figure 4-25. Horizontal Sweep.

Q1120 acts as a switch. When the transistor is on (switch closed), the flyback line is pulled to ground and a current, increasing with time, flows through the yoke winding. When Q1120 turns off (switch open), the yoke's inductance causes it to charge the flyback capacitor, C491, to approximately 1000V. At this point, the current reverses through the yoke and the flyback capacitor discharges.

When the capacitor has discharged, the voltage on the flyback line passes through zero and the damper diode turns on, clamping at near zero volts. At this point, forward current starts through the yoke again. Shortly after the damper diode starts conducting, the Flyback Base Driver turns on the flyback transistor.

The horizontal retrace period, during which the 1000V flyback pulse appears on the flyback line, is determined by the inductance of the yoke and the value of the flyback capacitor. It is typically 8 μs. The amplitude of the current through the horizontal deflection yoke is determined by the voltage output of the horizontal sweep supply and sets the length of the horizontal sweep on the screen.

Horizontal linearity correction is achieved with the "s" correction capacitor, C991, and the saturable reactor, L791. They modulate the supply voltage during horizontal sweep, giving the current ramp a slightly "s" shaped amplitude contour.

Flyback Pulse Regenerator

The Flyback Pulse Regenerator consists of a one-shot multivibrator. The 1000V flyback pulse, appearing on the collector of Q1102, is scaled down to about 5V to produce FLYBACK SENSE. This signal triggers the one-shot which, in turn, generates FLYBACK TRIGGER, a pulse of 4 μs width. FLYBACK TRIGGER is used by the Up-Down Ramp Generator, the High Voltage Inhibit circuit, and the Horizontal Ramp Generator.

Up-Down Ramp Generator

The Up-Down Ramp Generator receives FLYBACK TRIGGER. It produces a steep positive-going ramp during the time that the FLYBACK TRIGGER is high and a gently sloped, negative-going ramp while FLYBACK TRIGGER is low. The up-down ramp is passed to the phase comparator in the Horizontal Oscillator and Phase Comparator where it is compared to the delayed HSYNC-0 pulse.

Horizontal Sweep Supply and Picture Width Regulator

The Horizontal Sweep Supply and Picture Width Regulator supplies the voltage to drive the horizontal sweep. This supply feeds the horizontal sweep circuit through L661. The inductor, L661, serves to reduce feedback of high frequency components into the horizontal sweep supply.

The output of the supply, which is about 95V, is varied by the setting of the horizontal width adjustment (HSIZE, R871). It is also modulated slightly over the vertical sweep interval by the geometry correction waveform from the Geometry and Vertical Linearity circuit.

Horizontal Position Supply

The Horizontal Position Supply is a floating, DC current source, connected across the horizontal yoke winding. Its power inputs are +8.1VDC and —8.1VDC, both referenced to a floating ground.

This circuit produces a constant DC current which is set by the horizontal position control (HPOS, R775). The DC current through the yoke shifts the position of the raster on the screen. The center of the adjustment (R775) provides zero offset current. When the adjustment rotated, an offset current is produced which moves the raster to the right or the left. The output of the Horizontal Position Supply passes through L691 which isolates it from the flyback pulse.

Horizontal Ramp Generator

The Horizontal Ramp Generator produces an output voltage which varies linearly, with time, from about —5 volts to +5 volts. It is reset to the negative voltage on each FLYBACK TRIGGER received from the Flyback Regenerator Circuit. The voltage at the end of the ramp (HRAMP) remains fixed at about +5V. However, the voltage at the beginning of the ramp can be adjusted by the left-right symmetry adjustment (L.R.SYMMETRY, R413). The HRAMP signal is used by the geometry circuits for generating the geometry correction waveforms and by the switching network in the convergence circuits. It is also used to synchronize the voltage supply.

CONVERGENCE AND Z-AXIS BOARD, EXTERNAL CONVERGENCE BOARD

Convergence Circuits

Refer to Schematics 10-1, 10-2, 10-3, and 11-1 and to the Color Video Display Block Diagram, Figure 6-10.

Color CRT

The 4027A uses a crt which has three electron guns arranged in a "delta" configuration (see Figure 4-26).

The crt also has a pattern of phosphor dots which coat the inside of the crt's faceplate. The phosphor dots are arranged into sets of three dots, or "dot triads," in a "Delta" configuration similar to the electron gun arrangement. In each triad, there is one phosphor dot for each of the primary colors, red, green, and blue. Just behind the phosphor dots is a structure called the "shadow mask". The shadow mask has a pattern of holes, one for each set of three phosphor dots. The electron guns are set at such an angle that, in conjunction with the shadow mask, each electron beam strikes only one set of phosphor dots on the faceplate. Thus the "blue" electron gun illuminates only the blue phosphor dots, and so on.

The three electron guns can be driven to four different levels of beam current, causing the three phosphors to illuminate to four degrees of intensity. A range of colors is produced from the resulting mixtures of these three colors.

This system is subject to certain kinds of distortion (see Figure 4-27). As the electron beams are deflected across the screen, small corrective deflections must be applied independently to each of the three electron beams so that they converge properly. The convergence electronics produce dynamic corrective signals which are applied to a convergence yoke to insure that the red, blue, and green images coincide in all areas of the screen.

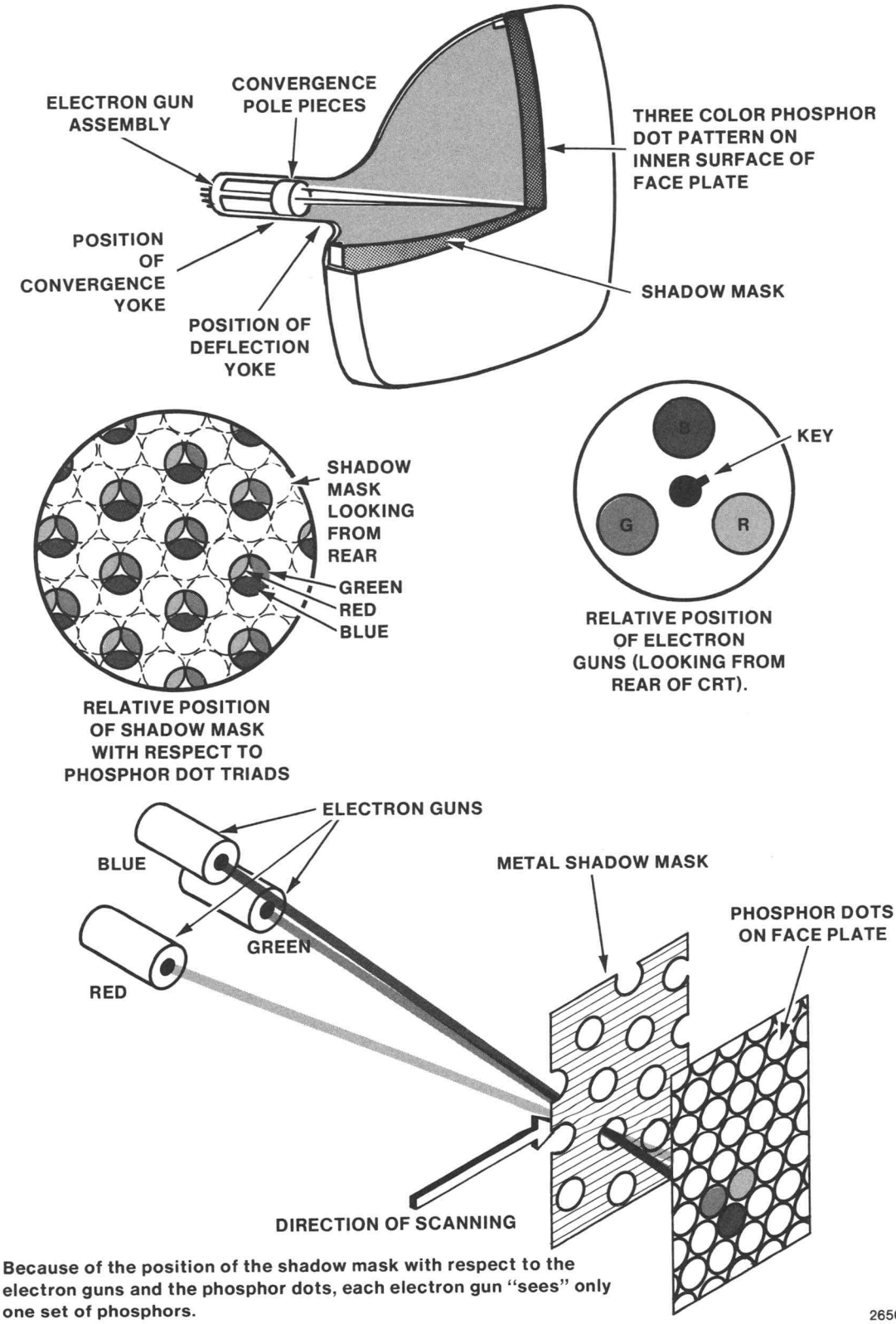


Figure 4-26. The Shadow Mask CRT.

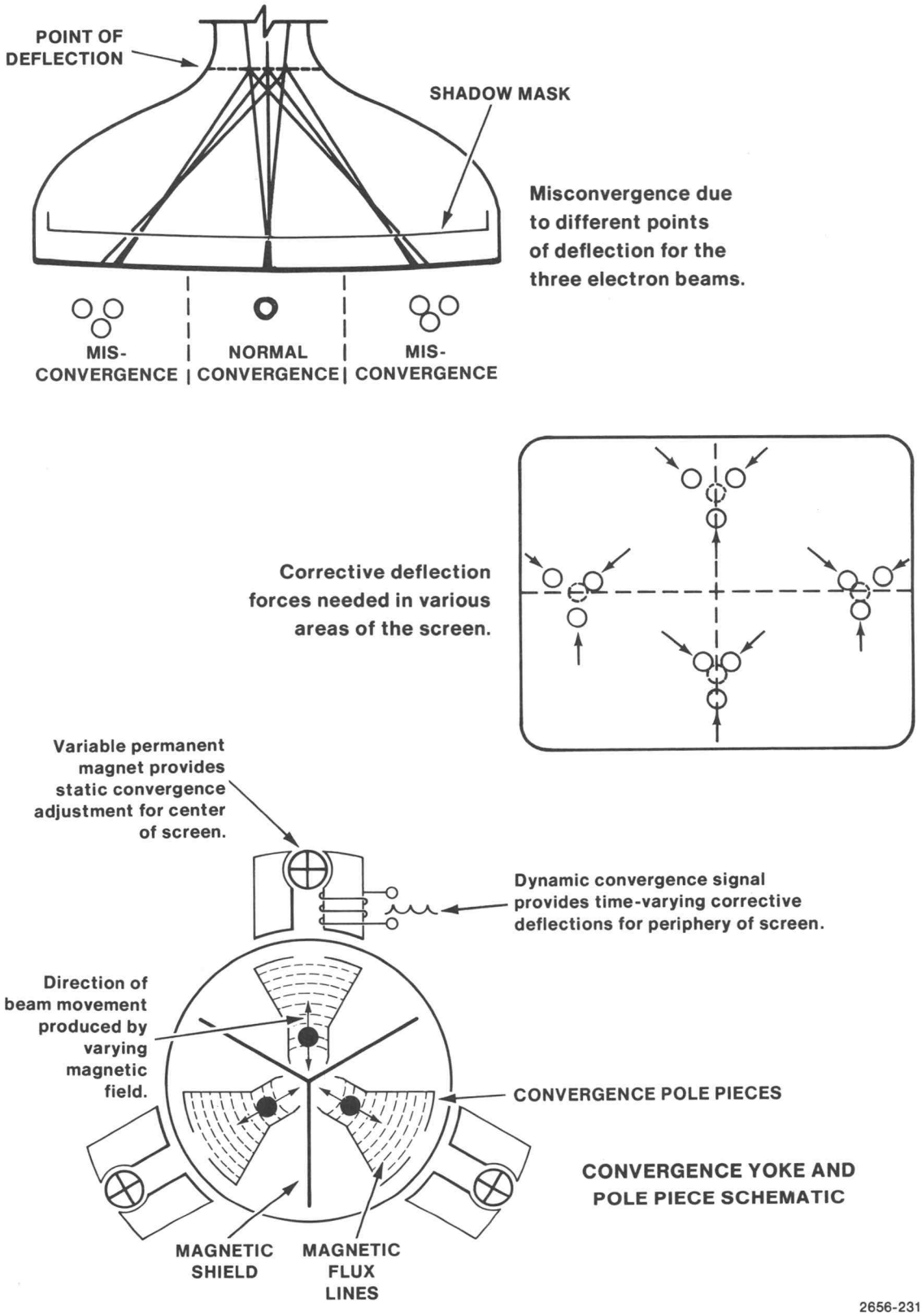


Figure 4-27. Convergence System.

Convergence Waveform Generators

See Schematic 10-1. Two voltage ramps, VRAMP (vertical) and HRAMP (horizontal) comprise the inputs to the convergence waveform generators. These signals are in phase with the vertical and horizontal sweep. They are used to produce three corrective signals: the vertical convergence signal, VCONV, the horizontal convergence signal, HCONV, and the corner convergence signal, CCONV.

HRAMP and VRAMP provide input to two inverting amplifiers (U531d and U542d, respectively). Horizontal and vertical balance adjustments (HORIZ BAL, R403 and VERT BAL, R501) allow these signals to be offset with respect to 0 Volts at the outputs of the amplifiers. The outputs of the amplifiers swing between approximately —4 volts and +4 volts, depending on the setting of the balance adjustments and they may be offset plus or minus 1 volt with respect to ground.

The adjusted HRAMP and VRAMP are sent to integrated circuits U541 and U451, respectively. These devices transform the two ramp signals to produce the horizontal and vertical corrective signals, HCONV and VCONV.

The horizontal amplitude adjustment (HORIZ AMP) sets the amplitude of HCONV which, from U541, is capacitively coupled to the noninverting side of U531b. This amplifier acts as a buffer. The horizontal offset adjustment (HOR OFF, R601) is used to set the bottom of the waveform at zero volts.

The VCONV waveform is capacitively coupled from U451 to the noninverting side of U452c. The vertical offset adjustment (VOFF, R503) is used to set the bottom of this waveform at 0 volts.

A third convergence waveform is generated by a differential amplifier, U531c, in conjunction with U451 and U541. The absolute value of VRAMP (ABVRAMP) appears at Pin 11 of U541. This is applied to Pin 12 of U541. The resultant output, at Pin 13 of U541, is a combination of ABVRAMP and HCONV. This signal is applied to the inverting input of U531.

ABVRAMP is applied to the noninverting input of U531c. Since U531c is a difference amplifier, the ABVRAMP component, which appears at both inputs, is cancelled out.

The output of U531c is the corner convergence waveform, CCONV. It resembles a series of HCONV waveforms that increase from minimum amplitude, while the electron beams are at the center of the screen, to maximum amplitude at the extremes of vertical deflection (top and bottom of the screen).

Waveform Sorting

The convergence waveforms go to the waveform sorting circuit where they are broken up into top and bottom, right and left, top right and top left, bottom right and bottom left components.

HRAMP and VRAMP drive comparators U551c and U551a, respectively. These two circuits are "zero-crossing" switches; they detect the zero crossing of the ramp signals and switch their outputs high or low accordingly. A filter network, C641 and R641, serves to minimize noise on VRAMP which might cause false switching.

As HRAMP crosses zero volts in the positive direction, U551c's output goes low and U551d's output switches high (to about 13V). Likewise, for VRAMP, U551a's output goes low and U551b's output goes high. These outputs control 10 analog switches which allow the three convergence waveforms coming from U531b, U531c, and U542c to pass or block them, depending on the state of their control gates. For example, when Pin 13 of U551d is high (beam at right side of screen), the signal present at Pin 1 of analog switch U562a can pass through to Pin 2. At the same time, since the voltage at Pin 14 must be low, the signal at Pin 4 of U562b is blocked from passing through to Pin 3.

External Convergence Board

See Schematic 11-1. The External Convergence board contains three summing amplifiers. The purpose of these amplifiers is to combine the various components (top, bottom, etc.) of the convergence waveforms to produce a composite dynamic convergence signal for each of the three convergence coils. There are nine potentiometers at the inputs to each of the three summing amplifiers. Eight of these potentiometers allow the proper amount of each of the eight sorted waveform components to be summed at the amplifier inputs. One potentiometer is an overall offset adjustment which is used to adjust the convergence at the center of the screen. In all, there is a dynamic convergence adjustment for each of the three beams in nine different areas of the screen.

Convergence Amplifiers

The composite convergence waveforms from the External Convergence board are fed to the Red, Blue and Green Convergence Amplifiers. These amplifiers drive the convergence coils located on the crt neck. They are identical inverting, complementary symmetry amplifiers employing negative feedback. The feedback samples are taken across a 1-ohm resistor in series with one end of each convergence coil. Then samples are applied through 1.2K ohm resistors to summing nodes at the inputs of the amplifiers. The Blue Convergence Amplifier receives an inverted signal (by way of U542d).

Z-Axis Circuits

Refer to Schematic 10-2 and Figure 6-10, the block diagram. The Z-Axis circuit controls the color and intensity of the display by converting digial information from the Display Controller into current levels in the three electron guns in the crt. In addition, the Contrast Control circuitry allows the relative intensity of the displayed picture to be varied from bright to very dim.

+100 Volt Regulator

The +100 Volt Regulator converts +110V from the main power supply into a clean and stable +100V level for use by the Z-Axis circuit. The circuit consists of a series-pass regulator, Q121, whose output is fed back to a two-transistor differential error amplifier (Q430 and Q431).

The feedback network is a resistive voltage divider connected from the +100V output to the —12V supply. The feedback voltage taken at the junction of the 12K ohm and 100K ohm resistors is approximately 0V. This is compared to reference ground in the differential amplifier. Any difference between the reference and the fed-back signal is amplified and applied to the series pass output in a way that corrects the voltage output. Incorporated in the +100V Regulator Circuit is a foldback transistor (Q432) which limits current in case the +100V output is accidentally grounded.

Digital-to-Analog Converters

The Digital-to-Analog Converters (DACs) comprise three major parts: the Input Decoders, the Converter Stages, and the Cascode Outputs.

Input Decoders. The Input Decoders are two-line to four-line, TTL data decoders. These decode the 2-bits of data for each color coming from the Display Controller to select one of four output lines. Three of these output lines are used to control the converter stages. One line, corresponding to black, is left unconnected.

Converter Stages. Three identical converters change the outputs from the 2-line to 4-line decoders into currents that drive the cascode output stages.

Each converter consists of six transistors; these are connected in three emitter-coupled pairs. One side of each pair, the current switch, is driven by one of the Input Decoders. The current switch controls the second side of the pair, the current source, which is connected (the bases of the transistors) to a voltage reference supplied by the Contrast Control. The switched currents through the current sources are summed where the three current source transistor connectors are connected together. This output current drives the cascode output stages.

The converter current is controlled by three factors: the output voltage of the contrast control, the one of the three current sources selected by the input decoder, and the setting of the contrast gain adjustment. The purpose of the contrast gain adjustment is to allow independent setting of the amount of drive to each crt gun. This is needed to compensate for non-uniform crt drive requirements. The contrast control varies the current through all three converters simultaneously. This controls the display brightness.

Cascode Output. Three power transistors form cascode circuits with the current source transistors in the Converter Stages. These transistors drive the crt cathodes. This, in turn, controls beam current and intensity. The output transistors' collectors are peaked with a T-coil to improve frequency response into their capacitive cathode load.

Contrast Control

The Contrast Control allows the intensity of the display to be varied. This circuit contains two emitter follower buffers which supply two voltage outputs. One buffer drives the bases of the current source transistors in the DAC and sets (in part) the current through these transistors. It also connects to the high side of the contrast gain set potentiometers (R101, R201, and R301). The input to this first buffer (Q441) is a fixed voltage taken from the top of the contrast control potentiometer, R1807.

The input to the second emitter follower buffer comes from the wiper of the contrast control. Therefore, the buffer provides a DC voltage which varies directly with the setting of the contrast control. This variable voltage is connected to the low end of the three contrast gain set potentiometers. As the contrast control is varied, the output of the contrast gain set circuit varies proportionately. This causes the current through the DAC current sources to vary, resulting in a change in cathode drive.

Contrast Gain Set

The contrast gain set circuit scales the variable voltage from the contrast control circuit. Each color has its own gain set so that the relative intensities of red, blue, and green can be balanced to give an accurate white.

Cursor Switch

The cursor switch is a level shift circuit that slightly lowers the variable output of the contrast control circuit. The switch is active when the CURSOR-0 line goes true. The slightly lowered output of the contrast circuit increases the current through the current sources, thereby increasing cathode drive and intensity. In this way, the cursor is slightly intensified compared to other information on the display.

HIGH VOLTAGE CONTROL AND HIGH VOLTAGE BOARDS

The High Voltage Control and High Voltage Boards are covered by Figure 6-10, the Color Video Display Block Diagram, and Schematic 13-1. The high voltage power supply consists of eight subcircuits on two circuit boards. These are the Gated Colpitts Oscillator, the Controlled Gain Amplifier, the Output Amplifier, the High Voltage Transformer, the Voltage Multiplier/Rectifier, the Grid 2 Sources, the Focus Voltage circuit, the Error Amplifier, and the Voltage-to-Current Converter.

Gated Colpitts Oscillator

The Gated Colpitts Oscillator provides a sine wave output which is converted into high voltage DC. The oscillator operates at about 31.5 kHz when it is free-running. In normal operation, it is locked to twice the horizontal scan frequency by injecting a small amount of signal from the horizontal ramp (HRAMP) into the base of the oscillator transistor. Locking is done to eliminate a ripple effect, due to low voltage power supply noise, which could be visible on the screen.

The oscillator may be gated off in two ways. First, it can be turned off by HV INHIBIT (High Voltage Inhibit) at the base of transistor Q155. HV INHIBIT is active when a fault condition occurs in the deflection circuits, or when the 12V or 5V supplies are lost. Secondly, it can be turned off by a signal which is derived from the voltage appearing on the primary of the High Voltage Transformer. This signal, which turns off Q155, provides overvoltage detection for the 21KV anode potential. Gating the oscillator off under a fault condition is important to protect the crt. High voltage cannot be generated when the oscillator is off.

Controlled Gain Amplifier

The Controlled Gain Amplifier consists of two parts: a pair of transistors, Q235a and Q235b, connected as an emitter-coupled differential amplifier, and a controlled current source (or current mirror) consisting of Q235d and Q235e.

The gain of the differential amplifier is set by the current flowing through the current source in the emitter circuit of the differential pair. This current, in turn, is controlled by a feedback signal which is proportional to the high voltage.

The output of the High Voltage Oscillator passes through the differential amplifier from the base of Q235a to the collector of Q235b. Since this output is used to generate the high voltage and, further, since the gain of the differential amplifier is set by negative feedback proportional to the high voltage output, the high voltage is stabilized.

High Voltage Output Amplifier

The high voltage output amplifier is a two-stage, capacitively coupled, power amplifier consisting of Q423 and Q1120. Q1120 drives the primary of T440 with an approximate 250-volt peak-to-peak sine wave of twice the horizontal sweep frequency.

High Voltage Transformer

The High Voltage Transformer (T440) steps up the 250-volt sine wave from the high voltage output amplifier to 7000 volts peak-to-peak. A tap on the secondary, at about 700 volts peak, is used by the Grid 2 sources.

To increase efficiency, the transformer primary is resonated with stray and fixed capacitance (C535 and C435) at approximately twice the horizontal sweep frequency.

Voltage Multiplier

The High Voltage Multiplier receives 7000V peak to peak from the High Voltage Transformer. The principal function of the multiplier is to rectify and to multiply the voltage by a factor of three. This produces the 21KVDC anode output. A second output from the multiplier (6Vdc) feeds into the DC error amplifier. A third output, approximately 5KVdc, goes to the focus voltage circuit.

Focus Supply

The focus voltage is, in part, a 5KVDC level from the voltage multiplier. The DYNAMIC FOCUS signal from the Deflection board is capacitively coupled to it. The resulting focus voltage is a 5KVDC level with an AC signal on it which dynamically focuses the electron beam as it scans the screen.

Grid 2 Supply

The output of the 700-volt high voltage transformer tap passes through a half-wave rectifier. This produces a DC voltage which is applied to tree variable resistors. The resistors are used to set the Grid 2 potentials for the three electron guns.

DC Error Amplifier

The Error Amplifier circuit detects variations in the high voltage output. It takes its input from the 6V trap on the voltage multiplier. This voltage is compared to a zener voltage reference by an operational amplifier, U115. The amplifier error output is coupled to the current source transistors in the Controlled Gain Amplifier through a voltage-to-current converter. The Error Amplifier varies the gain of the Controlled Gain Amplifier and, hence, the drive to the High Voltage Transformer.

Voltage to Current Converter

This nonlinear RC network transforms the voltage output of the Error Amplifier to the current input required by the current source in the Controlled Gain Amplifier. The values of resistance and capacitance used in this network are critical for establishing transient response and overall stability.

Degaussing Circuit

The purpose of degaussing is to disrupt weak magnetic dipoles in structures surrounding the crt. These weak magnetic dipoles tend to align themselves with stray magnetic fields such as the earth's field. Without degaussing, they can combine in strength to distort the deflection path of the electron beam and this may result in poor color purity on the crt.

The degaussing circuit consists of a 100-turn coil mounted on the chassis in front of the crt, a temperature sensitive resistor (thermistor), a capacitor, and a switch that connects to the primary AC voltage. When the coil is energized, 60-cycle-line voltage appears across the coil and the thermistor in series with the coil. The resultant AC field disrupts the residual magnetism around the front of the crt. As current heats the thermistor, its resistance increases, allowing less and less current to flow. This gradually diminishes the degaussing field.

LOW VOLTAGE POWER SUPPLY

The 4027A employs a high-efficiency, switching-type power supply. The supply is contained primarily on two circuit boards: the Inverter Board and the Power Supply Board. The circuits appear in Schematics 14-1, 2, and 3. Figure 6-11 is the block diagram for this circuitry.

In this power supply, the line voltage is first rectified and filtered. The rectified line voltage is then chopped, by an inverter, to produce a 20kHz alternating circuit in the primary of a small, high-frequency transformer. The 20kHz current in the transformer's secondary windings is used to produce several supply voltages.

Inverter Board and Other Primary Circuits

WARNING

Circuitry on the Inverter Board is directly connected to the AC line. An isolation transformer must be used when servicing this part of the terminal.

Refer to Schematic 14-1. AC power enters the terminal through an integral line filter, fuse holder, and line voltage selector assembly. A small transformer (T1220) connected to the line voltage selector serves four purposes: to supply power to run the control circuitry, to supply power for the pilot light in the power switch, to supply power for the fan, and to supply 115VAC to the degaussing coil circuit.

The rectifier, CR1220, is connected as a voltage doubler when the line voltage selector is in the 100VAC or 120VAC position. It is connected as a full-wave bridge when the selector is at 220V or 240V. Two filter capacitors, C1221 and C1222, smooth the rectified voltage.

As shown in Figure 4-28, 300 to 350VDC appears between the negative side of C1221 and the positive side of C1222. This is the DC supply to the switching transistors, Q101 and Q201.

Switched current flows alternately through Q101 and Q201, then through the primary of T441, T21 (the current sense transformer), C121, and finally, into or out of the common connection of the capacitors at Pin 3 of P229.

The switching transistors are driven by transformer T221 through an RC-diode network which aids the transformer in turning the transistors off and on. Another RC network (C111 and R11) helps keep the transistors in their safe operating area (prevents excessive power dissipation). The alternating current (approximately 20KHz) through the primary of T441 powers the various regulated supplies through several secondary windings.

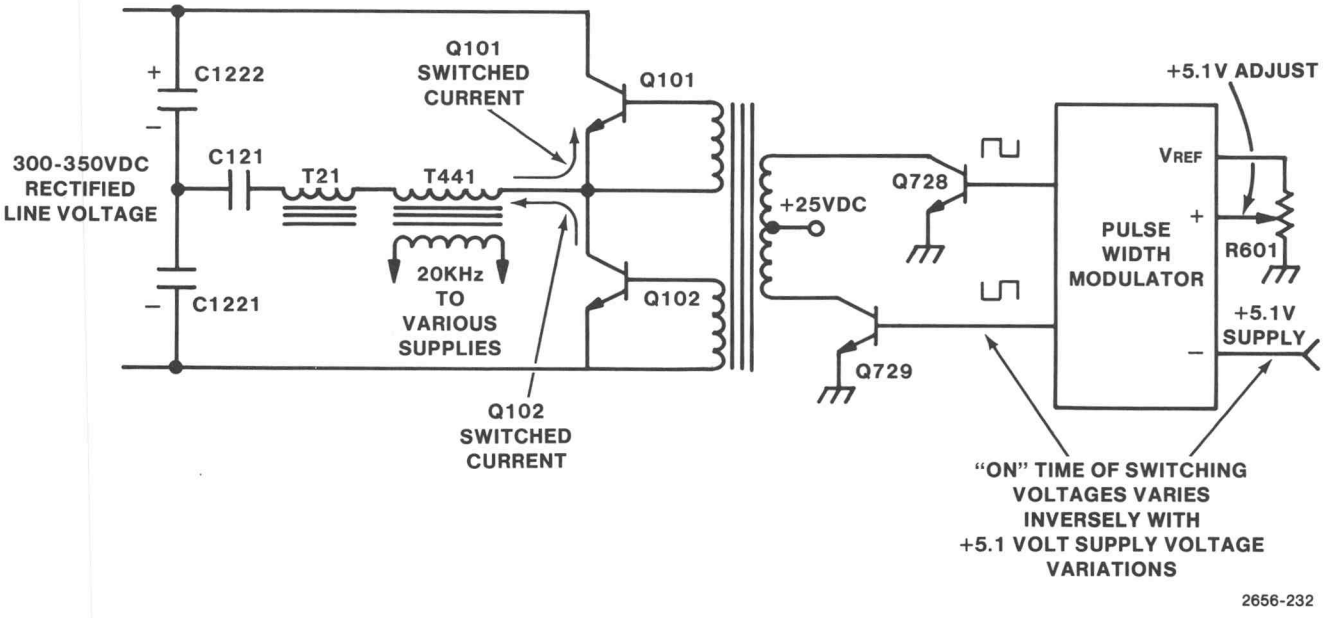


Figure 4-28. Low Voltage Power Supply Inverter, Simplified Schematic.

Pulse Width Modulator (PWM) and Control Circuits

Refer to Schematic 14-1. Most functions of the PWM and Control Circuits are accomplished by U611, a regulating, pulse-width modulator I.C. This device provides a voltage reference, oscillator, and current limiting and switching drive voltages.

The 5.1V sense line passes through a voltage divider into the inverting input (Pin 1) of U611. A reference voltage at Pin 16 of U611 is divided down by a voltage divider and appears at the noninverting input of U611. R601 is used to set the voltage of the 5V supply. Any deviation of the sensed voltage at the inverting input of U611 from the voltage set by R601 at the noninverting input causes U611 to change the "on" time of its switching outputs at Pins 11 and 14. If the +5v supply increases, U611 compensates by causing switching transistors Q101 and Q201 (by way of Q727, Q623 and T221) to turn on for a smaller amount of time. The +5V supply is the only one that is adjustable.

Pin 9 is the output of an internal voltage comparator which is used as a shut-down control point. Pin 9 is connected to a shut-down circuit whose output is Q751, and a power-up circuit whose output is Q717. Q717 keeps the switching outputs of U611 from turning on until sufficient voltage is available to drive the switching transistors.

The operating frequency of the oscillator is set by C711 and R615 to about 20kHz. PSYNC (Power Supply Sync), from the logic circuits, connects through a diode to Pin 3 of U611 to synchronize the power supply oscillator with the horizontal sweep. This prevents a "ripple" effect from appearing on the display.

The switching outputs of U611, Pins 11 and 14, are driven alternately; they drive Q623 and Q727 which, in turn, drive the base drive transformer, T221. Pins 12 and 13 supply a voltage that shuts off drive (through CR733 and CR731) in case of a spurious noise spike.

Current Limiting Circuit

Refer to Schematic 14-1. U611 contains a Current Limiting circuit which is used to suppress the switching outputs in case of an overload. The output of the current sensing transformer passes through two rectifiers (CR31 and C131) to supply a DC voltage which varies directly with the load current in the primary circuit. This is filtered and applied (by way of a voltage divider) to one of the current limiting inputs of U611.

Another current limiting input, at Pin 5, is connected through a voltage divider to the 5.1V supply. This input provides foldback current limiting. As the 5.1V supply voltage begins to drop, during current limiting, the output of the voltage divider also drops. This causes the current limit circuitry to allow even less current in the primary of T441. When a short circuit occurs, this foldback action allows very little current in the primary of T441.

Power Fail Detection

Refer to Schematic 14-1. The Power Fail Detect circuit senses the presence of the AC line voltage through T1220. CR737 and CR735 provide the input signal to the Power Fail Detect circuit. If power fails, C731 discharges faster than the filter capacitors in the power supply. The comparator (Q841 and Q845) detects this low line voltage. R833 is set to produce a low-line detection at about 100VAC when the line voltage selector is in the 120VAC position. When a low-line condition occurs, Q845 turns on Q751. This, in turn, shuts down the pulse-width modulator.

Over-Temperature Detection

Refer to Schematic 14-1. The Over-Temperature Detect circuit consists of a comparator (Q741 and Q748) and a temperature sensor RT455. When the temperature inside the terminal exceeds 80 degrees centigrade, the comparator shuts down the power supply by turning on Q751.

PWDN-0

Any time a shutdown condition occurs, Q857 is turned on. When Q857 switches on, a Power-Down warning (PWDN-0) is given to the Procssor approximately 2 ms before the Pulse-Width Modulator is shut down (PWDN-0 is connected to the reset line on the processor bus). On power-up, approximately 20 ms after the 5V supply has come up to voltage, PWDN-0 goes false. This initiates RESET functions in the terminal. See Figure 4-29.

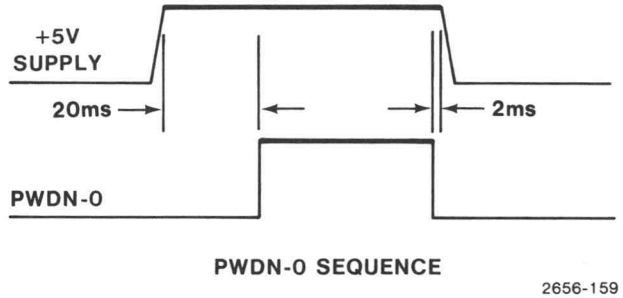


Figure 4-29. Power Up-Down Sequence.

5.1VDC Supply

Refer to Schematic 14-2. The 5.1V supply derives power from T441. This is rectified and filtered by a two-stage LC filter. The 5.1V supply contains a crowbar over-voltage protection circuit to prevent damage to the logic circuitry. When the 5.1V output goes over voltage (5.9 to 6.5V), Q155 turns on an SCR (Q1201). The SCR pulls the output down to about 1 volt. Regulation, as previously discussed, is by feedback through the 5.1V sense line to the Pulse Width Modulator circuit, U611.

Filament Supply

Refer to Schematic 14-2. The filament supply consists of a winding on T441, two rectifiers, an LC filter, a 1.5 amp fuse, and a dropping resistor (to give the proper voltage range). This supply is tied to power supply ground through a resistor, R757. In case of arcing in the crt, this allows the filament supply to float with reference to ground and prevents damage to the filaments.

+ and — 12 Volt Supplies

Refer to Schematic 14-2. The +12 volt and —12 volt supplies are connected to the same winding on T441. The +12V supply is referenced to the —12V supply, which is in turn referenced to the +5.1V supply.

The +12V supply uses a series-pass regulator transistor. The regulator is controlled by operational amplifier U111b which senses output voltage changes. Foldback current limiting is controlled by Q223 and Q225. The collector of Q225 is connected to the noninverting input of U111. This shuts down the +12V supply in an overcurrent condition. Q131 and its associated components provide crowbar overvoltage protection. The —12V supply is similar to the +12V supply except that the transistors are of the opposite polarity (NPN) and the —12V supply does not have a crowbar circuit.

+ and — 8.1 Volt Supplies

Refer to Schematic 14-2. The + and — 8.1V supplies are floating. The ground reference is connected to a point in the deflection circuit and floats between 70V and 90V above power supply ground. The capacitor associated with the LC filter is located on the Deflection board. The 2.7 ohm resistors provide short circuit protection.

—5 Volt Supply

Refer to Schematic 14-3. The —5V supply incorporates a series-pass regulator, Q181, controlled by an operational amplifier, U271. Foldback current limiting is provided by Q271 and Q273 and their associated circuitry.

+ and — 30 Volt Supplies

Refer to Schematic 14-3. The + and — 30V supplies operate from the same winding on T441. The series pass element for the +30V power supply is a Darlington transistor, Q121. It is controlled by operational amplifier U111a and a foldback current-limiting circuit composed of Q313 and Q311 and their associated circuitry. The —30V supply is similar except that the transistors are of the opposite polarity.

+110 Volt Supply

Refer to Schematic 14-3. The +110V supply is unregulated in normal operation, and it has a 1-amp fuse for short circuit protection. This supply contains a shunt regulator which is activated only when the display and high voltage circuits are disabled and do not draw sufficient current from the supply. The shunt regulator keeps the output voltage from exceeding 160 volts. This condition will not occur during normal operation, but it might occur during servicing.

OPTION 1, HALF DUPLEX

The Half Duplex option consists of one ROM and a Half Duplex cable. The ROM contains firmware which adds the DUPLEX and DISCONNECT commands to the terminal's command set. For more information on these commands, refer to the 4027A Programmer's Reference manual. The circuitry involved in half-duplex operation is covered under the Deluxe Communications Board circuit descriptions.

The Half Duplex ROM is installed in U35 on the Processor board. See Volume 2 of this service manual for additional information on the installation of this option.

OPTION 2, CURRENT LOOP INTERFACE

The Current Loop Interface allows the 4027A to communicate with an external device by means of a current loop rather than the standard RS-232 interface. Schematic 15-1 covers the Current Loop Interface circuit board.

The Current Loop Interface lies between the host computer or modem and the host port of the Deluxe Communications board.

The Current Loop Interface is connected to the host computer or modem with a "current loop" cable. This cable has conductors for two independent circuits, or loops. One circuit (the transmit loop) carries data to the computer; the other (the receive loop) carries data from the computer to the terminal.

There are two modes of operation for the Current Loop Interface: Active mode and Passive mode. Figure 4-30 is a simplified schematic diagram of the Current Loop Interface with both receive and transmit loops set to Passive mode (setting the Active/Passive jumpers is described in Volume 2 of this manual.) In Passive mode, the Current Loop Interface relies on an external device — the computer or modem — to provide a 20-ma current source for the loop.

The current in the receive loop controls the Current Loop Interface's receiver circuitry. The receiver circuit drives the terminal's RDATA line. Current flowing in the loop is interpreted as a "mark," or binary one; the absence of current is interpreted as a "space," or binary zero. Binary ones send RDATA negative; binary zeroes send RDATA positive.

Likewise, the TDATA signal from the terminal controls the Current Loop Interface's transmitter circuitry, which in turn keys the transmit loop. A negative voltage on TDATA is interpreted as a mark or binary one, causing current to flow in the transmit loop. A positive voltage on TDATA is interpreted as a space or binary zero, shutting off the flow of current.

Note that with the Current Loop Interface in Passive mode, conventional current (a flow of positive charges) enters the Current Loop Interface through the T+ lead and leaves it through the T- lead. Likewise, conventional current enters the R+ lead and leaves the R- lead.

Figure 4-31 shows the Current Loop Interface set to Active mode in both transmit and receive loops. Compared to Passive mode, there are only two differences:

- It is the Current Loop Interface, rather than the computer (or modem) which provides the source of current in the loops.
- Conventional current (a flow of positive charges) leaves the T+ lead and returns by the T- lead. Likewise, conventional current leaves the R+ lead and returns by the R- lead.

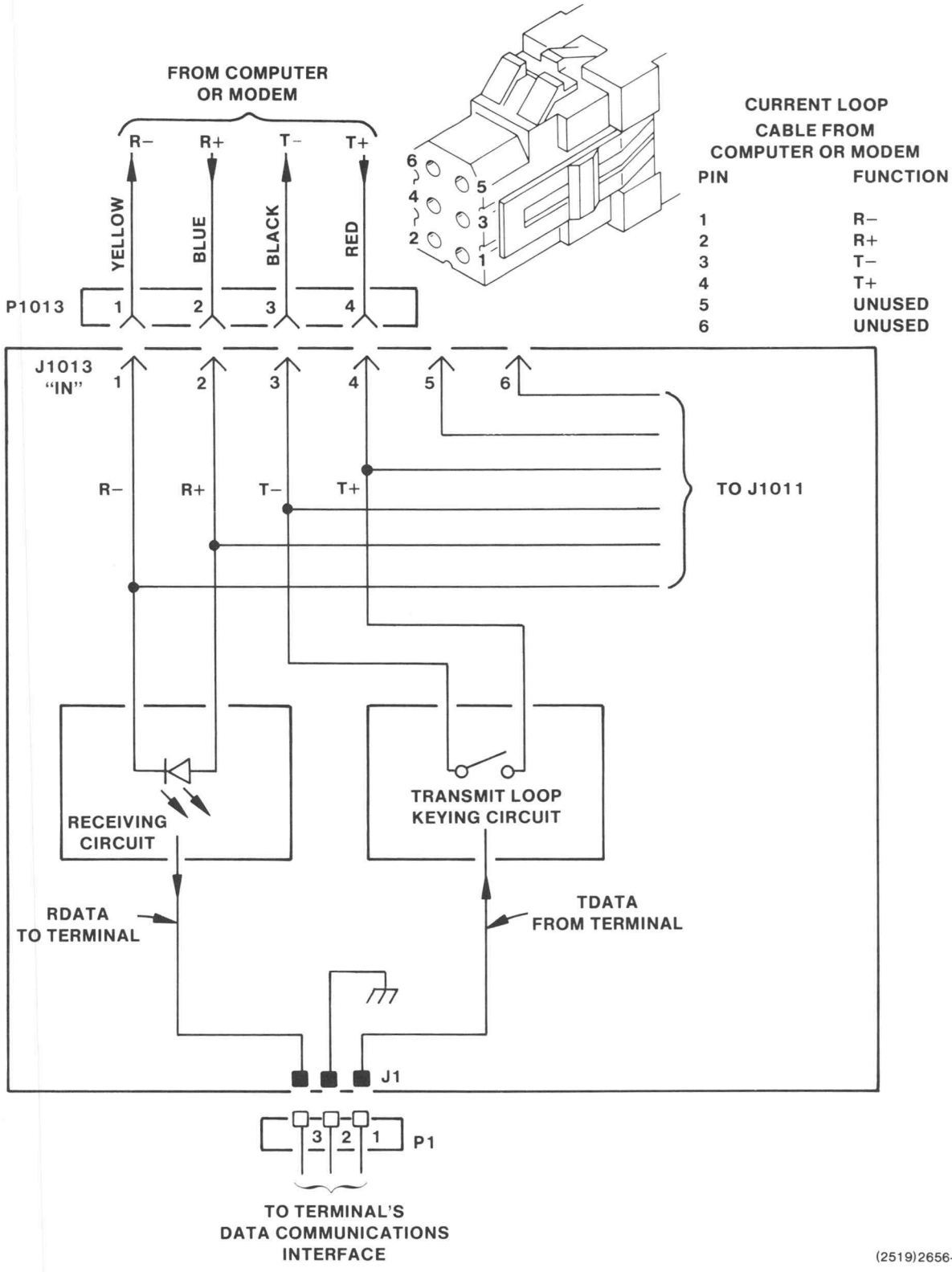


Figure 4-30. Option 2 in Passive Mode.

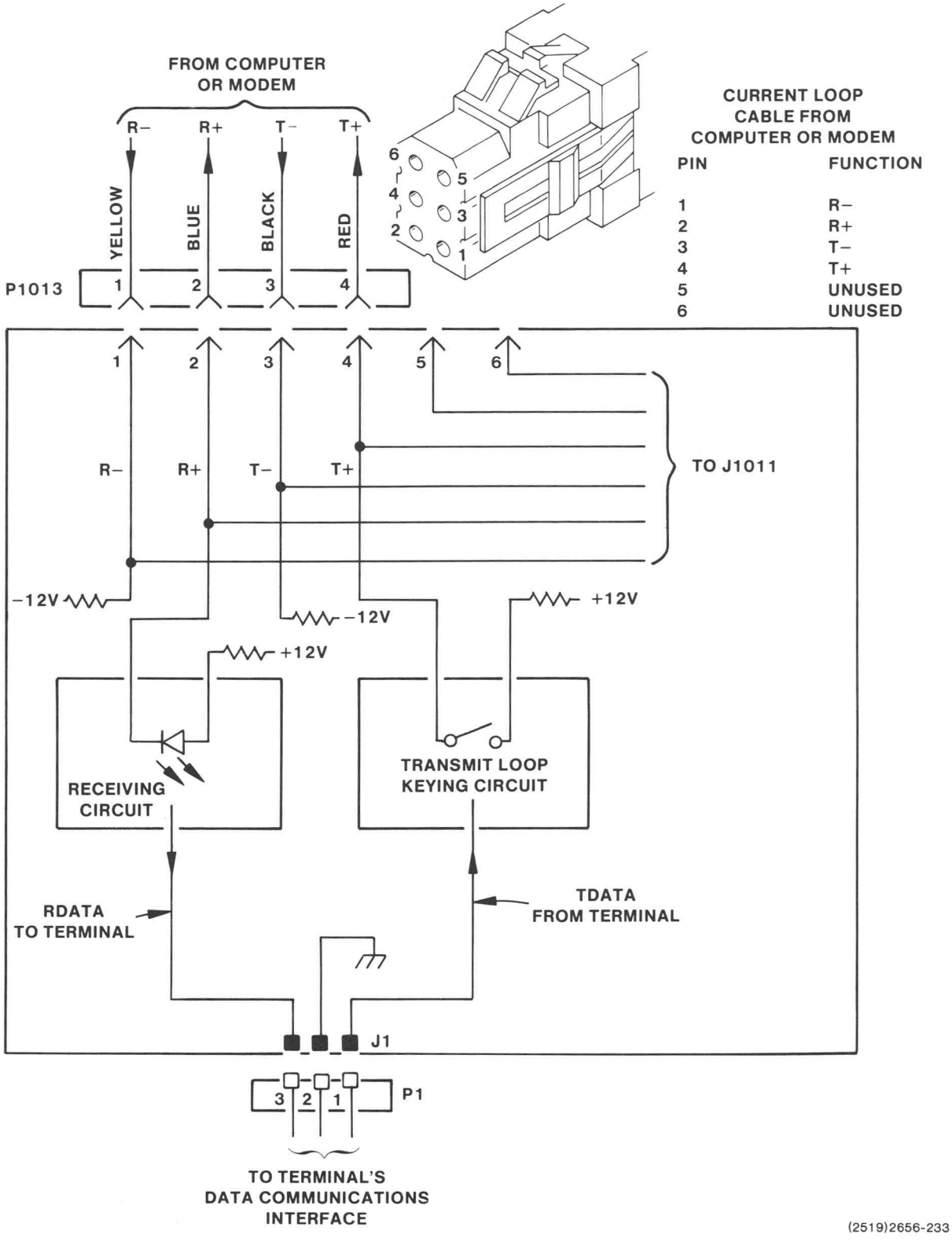


Figure 4-31. Option 2 in Active Mode.

OPTION 3, RS-232 PERIPHERAL INTERFACE

The RS-232 Peripheral Interface provides a serial data communications port. Contained on one etched circuit board, the circuitry is designed around a large-scale integrated circuit, the 8251 USART (Universal Synchronous/Asynchronous Receiver/Transmitter). In addition to serial input and output, the circuit provides flag inputs and outputs for "handshaking" purposes, an external clock input, and a transmitter clock output. Firmware, which drives the Tektronix 4641 and 4642 Printers through the RS-232 Peripheral Interface, is supplied as Option 36 (Peripheral ROMS).

NOTE

The receive capabilities of the RS-232 Peripheral Interface Board are not used by the Option 36 firmware. Therefore, it can only be used to transmit data to a printer or a similar device. Although the receiver circuits are not used, some circuit description of this part of the board is included for reference.

The following points summarize the data handling characteristics of the RS-232 Peripheral Interface:

- Baud Rates: 75 to 9600, jumper selected; an external clock may be selected (see note below).
- Inputs: Serial data; two handshaking flags; external clock.
- Outputs: Serial data; two handshaking flags; transmit baud rate clock.
- Interrupts: Transmitter ready; receiver ready; flag status change; priority level jumper selectable (0 to 7).
- Data Format: The USART may be programmed to support several synchronous and asynchronous formats (see note).

NOTE

As set up by 4027A Option 36 (ROMs containing firmware for driving the TEKTRONIX 4642 Printer), the RS-232 Peripheral Interface provides asynchronous communications with eight data bits per character, no parity and two stop bits per character. The external clock must be 16 times the desired baud rate.

The RS-232 Peripheral Interface Board's circuitry appears in Schematics 16-1 and 16-2. The block diagram for the board is Figure 6-12.

The USART

The 8251 USART performs the principal function of the RS-232 Peripheral interface; converting parallel data bytes from the Processor into a serial bit stream. In addition, this device performs a number of coordinating functions which allow data to be transferred in an orderly fashion.

As used in the RS-232 Peripheral Interface, the USART is divided into four functional blocks: the data bus buffer, the receiver, the transmitter, and the read/write control logic (see Figure 4-32).

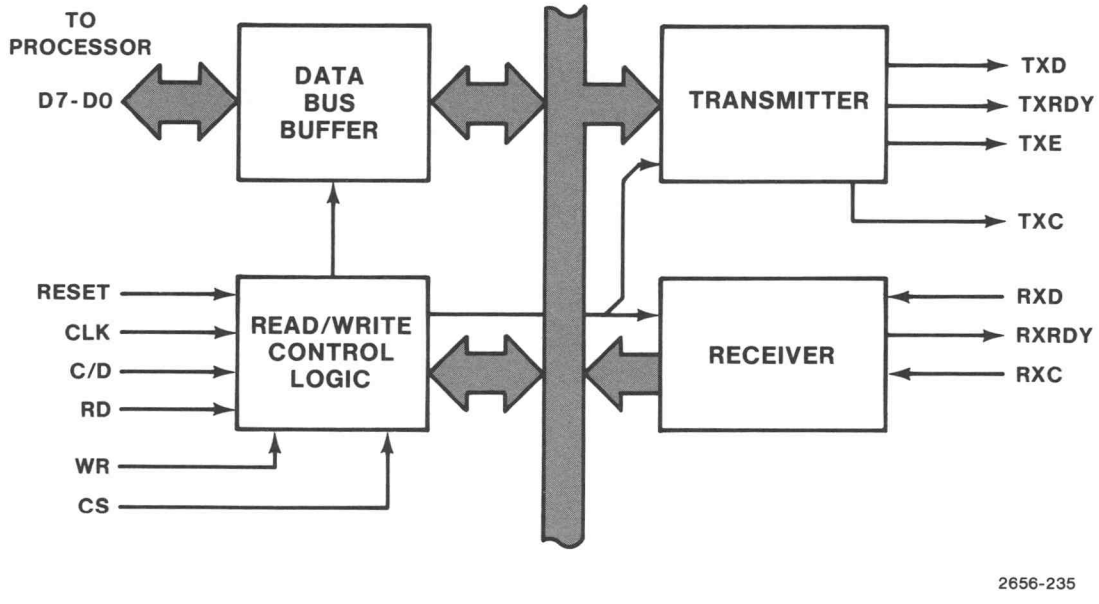


Figure 4-32. 8251 USART Functional Blocks.

The data bus buffer transfers data between the USART's internal data bus and the circuit board's data bus. These data bytes may be receiver or transmitter data or they can be commands to the USART or status information from the USART.

The read/write control logic coordinates routing and storage of data within the USART. It has a number of inputs:

- RESET. When RESET goes high, the 8251 enters an "idle" state. This prepares it to receive a set of control bytes from the processor which determine its mode of operation. Any time the RESET-0 line on the Mother board is pulled low, the USART gets a RESET.

- CLK (Clock) is used for timing various functions within the USART. The 2-MHz clock, LCLK-1 from the processor, provides this input.

- WR-0 (Write) informs the USART that information is to be written into it.

- RD-0 (Read) informs the USART that information is to be read from it.

- C/D-0 (Control/Data) in conjunction with WR-0 and RD-0, tells the 8251 whether the data to be transferred is character, control, or status information.

- CS-0 (Chip Select) must be asserted for reading or writing to take place in the USART.

Table 4-9 explains how WR-0, RD-0, C/D-0, and CS-0 work together to control data flow to and from the USART.

Table 4-9

USART I/O TRUTH TABLE

Signal State				Function
C/D-0	RD-0	WR-0	CS-0	
0	0	1	0	receiver data to data bus
0	1	0	0	data bus to transmitter
1	0	1	0	USART status to data bus
1	1	0	0	data bus to USART control

The transmitter accepts data bytes in parallel, then shifts them out bit-serially from the TxD output. It provides the following inputs and outputs:

- TxD (Transmitter Data) is at a high state while the transmitter has no data to send. As soon as the transmitter receives a character, TxD is pulled low for one bit period to produce a start bit. Then the data bits are transmitted. After the data bits have been transmitted, the line is held high for two bit periods to produce the stop bits.

- TxRDY (Transmitter Ready) signals that the transmitter is ready to accept a new data character. It is used to generate an interrupt and can also be read through Bit 1 of the board status word.

- TxE (Transmitter Empty) signals that the transmitter buffer is empty. This can be read through Bit 6 of the board status word.

- TxC (Transmitter Clock) controls the rate at which the data is transmitted. The falling edge of TxC shifts the data out of TxD.

The receiver accepts bit-serial characters through the RxD input and assembles them into eight-bit bytes. While waiting for incoming data, RxD is at a logic high state. When RxD goes low, the receiver waits one half bit period, then samples the input. If RxD is still low, the receiver interprets this as a valid start bit. The receiver then loads in the serial until an entire character has been assembled into the receiver buffer.

The receiver "knows" how many bits to assemble into the buffer because this is set up in the mode instruction to the USART when it is initialized. Once the specified number of data bits have been received, the receiver looks for the proper number of stop bits. If the stop bits are seen, the receiver signals that it has a character ready for processing. Failure to detect stop bits results in a framing error flag within the USART status word. The receiver has the following additional input and output:

- RxRDY (Receiver Ready) indicates that there is a character in the receiver buffer. It can be read through bit 3 of the board's status word.

- RxC (Receiver Clock) determines the rate at which data is shifted into RxD. Data is sampled on the rising edge of RxC.

Address Decoder

Refer to Schematic 16-1. Nine 74LS266 Exclusive-OR gates make up the address decoder. These gates have open-collector outputs which are connected together to form a wire-AND function. All of these outputs must go high to produce the signal MYADR-1.

The board's I/O Address is determined by the presence or absence of jumpers at one of the inputs to eight of the Exclusive-OR gates. If a jumper is installed, the input is connected to ground. Otherwise, the input is pulled up to 5 volts through a resistor. The other inputs to these gates are connected to the address bus lines BA2 through BA9. If the state of BA2 through BA9 matches the state of the jumpered inputs, the outputs of these eight gates can go high. A ninth gate is connected to IOADR-1. Its output can go high any time IOADR-1 is active. IOADR-1 is high any time the Processor addresses a location between X'0800' and X'0BFF'. The jumpers can be used to select a base address within this range.

In the 4027A, jumpers to ground are installed at positions A2, A3, A4, A5, A8, and A9. Therefore, the outputs of all the Exclusive-OR gates can go high (thereby asserting MYADR-1) at business addresses X'08C0', X'08C1', X'08C2', and X'08C3'. These are the RS-232 Peripheral Interface's status word, flag word, I/O data word, and USART word, respectively. The two low-order bits of the bus address are decoded by the I/O logic to select these four locations. Table 4-10 lists the purposes of the bits in the various I/O words.

Table 4-10

RS-232 PERIPHERAL INTERFACE I/O ADDRESS MAP

Bus Address	Bit	Description
X'08C0'		Status Word
	0	1=Transmitter Ready Interrupt enabled (write).
	1	1=Transmitter Ready (read only).
	2	1=Receiver Interrupt enabled (write only).
	3	1=Receiver Ready (read only).
	4	1=Flag Status Change Interrupt enabled (write).
	5	1=Flag Status Change (read only).
	6	1=Transmitter Empty (read only).
	7	1=Interrupt Condition Present (read only).
X'08C1'		Flag Word
	0	not used
	1	not used
	2	INFLAG1, 1=space, 0=mark (read only).
	3	INFLAG2, 1=space, 0=mark (read only).
	4	OUTFLAG1, 1=space, 0=mark (read/write).
	5	not used
	6	not used
	7	OUTFLAG2, 1=space, 0=mark (read/write).
X'08C2'		Data I/O Word
	0 thru 7	The processor reads and writes data characters through this address.
X'08C3'		USART Status and Programming
	0 thru 7	The processor writes mode and command instructions to the USART or reads the USART's status through this address (see Initialization below or the USART manufacturer's literature for additional information).

I/O LOGIC

Refer to Schematic 16-1. The I/O logic consists of several gates and a 74LS155 multiplexer which examines MYADR-1, BA0, BA1, WRITE-0,and READ-0 to determine where read or write operations are to be performed on the circuit board. It provides several outputs to control these operations:

- STATUSWR-0 selects the status word write register.
- USARTADR-0 is the chip select signal for the USART. This enables reading or writing in the USART.
- FLAGWR-0 enables writing into the flag write latch.
- STATUSRD-0 enables the status word read driver.
- MYREAD-0 controls the direction of data transfer through the bus transceiver.

See Table 4-11 which explains the conditions under which these output functions are active.

Table 4-11

RS-232 PERIPHERAL INTERFACE
I/O LOGIC TRUTH TABLE

READ-0	WRITE-0	BA0-0	BA1-0	MYADR-1	Function
1	0	0	0	1	STATUSWR-0
X	X	X	1	1	USARTADR-0
1	0	1	0	1	FLAGWR-0
0	1	1	0	1	FLAGRD-0
0	1	0	0	1	STATUSRD-0
0	X	X	X	1	MYREAD-0

X = irrelevant

Status Word Write Register

Refer to Schematic 16-2. The Status Word Write Register consists of a 74LS174 D-type flip-flop. Initially, the register is cleared by the RESET-0 line. When a write operation takes place at X'0800', STATUSWR-0 causes the content of data bus lines D0, D2, and D4 to be loaded into the register (see the I/O address map, Table 4-10).

Status Word Read Driver

The Status Word Read Driver (Schematic 16-2) consists of a 74LS367 Hex Bus Driver. Whenever a read operation takes place at X'08C0', STATUSRD-0 enables the 74LS367 causing several status bits to appear on the data bus (see the I/O address map, Table 4-10.)

Flag Write Latch

Refer to Schematic 16-2. The Flag Write Latch is one 74LS74 Dual D-type flip-flop. FLAGWR-0 causes the state of bus lines D4 and D7 to be stored in the latch when location X'08C1' is addressed. The Q-0 outputs of the flip-flops go to two RS-232 drivers. These provide flag signals, OUTFLAG1 and OUTFLAG2, for external devices. The Q-0 outputs can also be read through the Flag Read Driver.

Flag Read Driver

The Flag Read Driver consists of a 74LS368 Bus Driver. As a read operation is performed at X'08C1', FLAGRD-0 enables the 74LS368 causing the state of the flags to appear on the bus (see Table 4-10.)

Flag Status Change Detector

Refer to Schematic 16-2. The Flag Status Change Detector consists of several inverters, a couple of Exclusive-OR gates, and a flip-flop. When a change takes place in INFLAG1 or INFLAG2, this change (after passing through an RS-232 receiver) is seen at the inputs of an Exclusive-OR gate. The change is seen immediately at one input, but is delayed in getting to the other input because it must pass through an inverter, charge a capacitor, then pass through three more inverters on its way to the input. The Exclusive-OR sees a difference between its inputs causing it to pull its output low. When this occurs, the Q output of the 74LS74 is set high. This causes a flag status change interrupt to occur if this has been enabled. The state of the flip-flop can be read through the status word. It is automatically cleared upon being read (by STATUSRD-0) or when a system reset occurs (by RESET-0).

Operation With 4027A Option 36

4027A Option 36 consists of read-only memory. It contains firmware routines which the Processor uses in transmitting characters to the Tektronix 4642 and 4641 Printers through the RS-232 Peripheral Interface.

Upon power-up or RESET, Option 36 firmware causes the Processor to do the following:

1. Read location X'08C0' (board status word) to see if the board is installed (reading X'FF' at this location means the board is not installed.)

2. Store X'90' at X'08C1' (flag word). This sets OUTFLAG1 and OUTFLAG2 to the spacing (high) state.

3. Reset the USART. The Processor does this by:

   A. Writing 0 at X'08C3' (USART control and status).

   B. Waiting 32 microseconds.

   C. Doing A and B two more times.

   D. Writing X'40' at X'08C3'. This sets the INTERNAL RESET bit in the USART's status register and prepares it to receive the mode instruction.

4. Write X'CE' at location X'08C3'. This is the mode instruction. It sets up the USART for asynchronous communications with eight data bits per character, no parity, and two stop bits.

5. Wait 32 microseconds.

6. Write X'15' at X'08C3'. This is a command instruction. It resets all of the USART's error flags and enables its receiver and transmitter.

In transmitting characters to the printer:

1. The Processor gets a character from the printer queue (transmit buffer) and writes it at X'08C2' (the USART data port).

2. If this character is a control character such as a carriage return or line feed which requires a delay for mechanical action to take place, the Processor does one of the following (as determined by the terminal's SET command):

   A. Waits for Data Terminal Ready (DTR) from the printer. (This mode is used with the Tektronix 4642 Printer.) DTR is connected to the INFLAG1 input. While waiting for DTR, the processor enables line status change interrupts. As soon as a line status change interrupt occurs, it tests the flag word to see if INFLAG1 has gone true. If DTR is true, then Output interrupts are enabled and line status change interrupts are disabled.

   B. Goes into a time delay routine. (This mode is used with the Tektronix 4641 Printer.) The length of the delay is controlled by the SET command (which see). Its purpose is to allow the printer sufficient time to do a carriage return/line feed.

3. If the printer queue is empty, the Processor disables interrupts. (Input Interupts are always disabled because Option 36 does provide for receiving characters.)

OPTION 4, GPIB PERIPHERAL INTERFACE

The GPIB Peripheral Interface board provides the circuitry to interface the 4027A with the GPIB (General Purpose Interface Bus). The GPIB firmware which controls this interface is contained in the Option 36 Peripherals ROMs.

The circuitry is quite versatile; it can perform any standard GPIB function. However, the firmware is limited; it only provides for communication with the TEKTRONIX 4924 Digital Cartridge Tape Drive and the TEKTRONIX 4662 Interactive Digital Plotter.

NOTE

Volume 2 of this Service Manual contains installation information on Option 4.

About the GPIB

The GPIB is a standard interface for programmable instrumentation, defined in IEEE Standard 488-1975. The following description summarizes the pertinent parts of that standard.

Talkers, Listeners, and Controllers

There are three types of GPIB devices: **talkers, listeners,** and **controllers**. In any particular data transfer, the talker is the device which sends the data, and the listeners are the devices receiving it. The controller supervises the data transfer; it determines which device will talk and which will listen. (The Option 36 firmware requires that the 4027A be the only controller on the GPIB.)

GPIB Interface

The actual connection between terminal and peripheral devices is through a GPIB cable. This is a shielded, 24-wire cable which attaches to a standard connector (Figure 4-33) on the peripheral devices. Eight of the cable's wires are grounds; the other sixteen are logically divided into three distinct groups: a data bus, a management bus, and a transfer bus.

All devices on the GPIB are connected in parallel, and all GPIB signal lines are active low, passive high. A line is low if any GPIB device pulls it low (i.e., to ground) and high only if all devices let it float to a TTL high (i.e., +3.4V). In other words, devices on the GPIB are connected in a "wired-OR" configuration.

Data Bus. The GPIB data bus contains eight bidirectional signal lines. One byte of information (eight bits) is transferred over the bus at a time. DI01 (Data In-Out bit 1) is the least significant bit, and DI08 the most significant bit. Each byte represents a device address, a universal command, or a datum. (Device addresses and universal commands are distinguished from data by having the ATN line — in the management bus — activated while they are sent. With ATN asserted, certain bytes are reserved for universal commands and others for device addresses.)

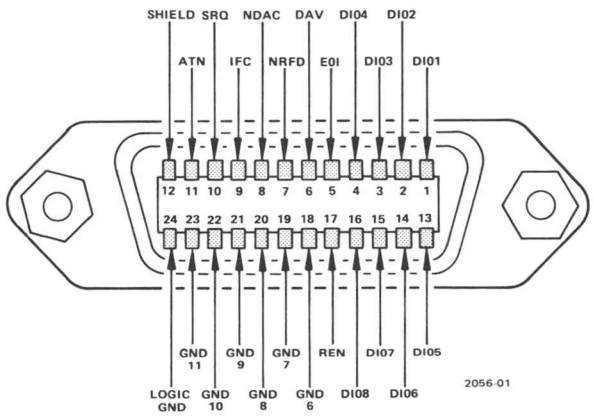


Figure 4-33. GPIB Connector.

Management Bus. The GPIB management bus consists of five signal lines for controlling data transfers. The lines are:

- **ATN (Attention).** The GPIB controller (that is, the 4027A) sends the ATN signal when assigning devices as listeners or talkers (sending device addresses) or when giving commands simultaneously to all devices on the bus (sending universal commands). Only device addresses and control messages can be transferred over the data bus when ATN is true. After ATN goes false (high), only those devices assigned as listeners and talkers can take part in the data transfer.

- **SRQ (Service Request).** Any device on the bus can request the controller's attention by sending SRQ active low.

- **IFC (Interface Clear).** The GPIB controller may send IFC true to put all devices on the GPIB in a known quiescent state. (The 4027A does this during power-up or RESET.)

- **REN (Remote Enable).** The REN signal is used in some GPIB applications to transfer devices from manual operation to remote control.

- **EOI (End of Identify).** Talkers may use the EOI signal to mark the end of a data transfer. (The talker sends EOI while simultaneously sending the last data byte in the sequence.)

Transfer Bus. Each time a byte is transferred over the data bus, the talker and listeners execute a handshake sequence using the following transfer bus signal lines:

- **NRFT (Not Ready For Data).** A low NRFD signal means that one or more assigned listeners are not ready to receive the next byte. When all the listeners have released NRFD, the NRFD line goes high. This tells the talker that it may place the next byte on the data bus.

- **DAV (Data Valid).** The talker sends DAV low shortly after placing a valid byte on the data bus. A true (low) DAV signal tells each listener to capture the data presented on the data bus. The talker is inhibited from sending DAV when NRFD is low.

- **NDAC (Not Data Accepted).** The NDAC signal is held low by each listener until it has captured the byte currently presented on the data bus. When all listeners have captured the byte, NDAC goes high. This tells the talker that it may remove the byte from the data bus.

Handshake Sequence

Figure 4-34 shows the transfer bus "handshaking" that regulates the exchange of data bytes on the data bus.

Initially, the listeners hold NDAC (Not Data Accepted) low, and the talker leaves DAV (Data Valid) high. One or more of the listeners may be holding NRFD (Not Ready For Data) low, indicating that it is not yet ready to accept a data byte.

When all listeners are ready for data, NRFD goes high. The talker then places a data byte on the data bus, waits briefly for the data to settle, and pulls DAV low. The low DAV signal indicates that valid data is available on the data bus.

The listeners then capture the data. Before beginning to accept the byte, each listener pulls NRFD low, indicating that it is not ready for the talker to place another byte on the data bus. Then the listeners read the data, and when done, release NDAC. When the slowest listener has captured the data, NDAC goes high; this tells the talkers that all listeners have received the byte.

The talker then releases the DAV line and changes the data byte on the data bus. the listeners, sensing DAV going high, pull down NDAC, preparing for the next data byte.

The process then repeats for successive data bytes.

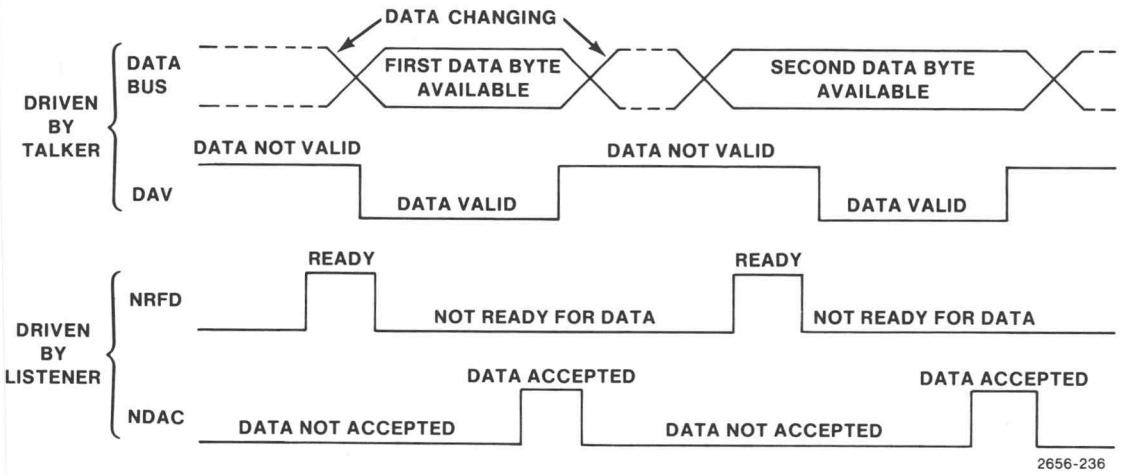


Figure 4-34. GPIB Transfer Bus Handshake Sequence.

GPIB Protocol: Addressing Devices for a Data Transfer

To cause a data transfer to occur, the 4027A (functioning as the GPIB controller) does the following:

1. First, it asserts ATN. This tells all other devices on the bus to stop whatever they are doing and pay attention to the bytes which the 4027A is about to send. While ATN is asserted, all devices on the bus must participate as "acceptors" in the data transfer handshaking. Each device examines the bytes it accepts, watching for its own **primary talk address** or **primary listen address**.

2. Next, the 4027A assigns one of the devices on the bus to be a listener. It does this by sending that device's primary listen address. (If the 4027A is to be the listener, it addresses itself internally, without sending its own listen address on the bus.)

3. Next, the 4027A assigns one of the devices to be a talker: it sends that device's primary talk address over the GPIB. (If the 4027A is to be the talker, it addresses itself internally, without sending its own talk address on the bus.)

4. The 4027A releases ATN. Only those devices which have been addressed as talker or listener are now permitted to take part in the data transfer.

   The device just addressed as talker then starts sending data bytes over the bus. The device (usually there is only one) addressed as listener receives the data. The data transfer is governed by the three-wire handshaking described earlier; the talker drives the DAV line and the listener drives NRFD and NDAC.

5. As the talker sends the last byte in the data sequence, it simultaneously drives the EOI (End of Identify) line low. This signals the 4027A that the data transfer is finished.

6. After the last data byte has been transferred, the 4027A pulls the ATN line low. While holding ATN low (true), it sends two bytes over the bus: the UNTALK and UNLISTEN bytes. These cause the devices currently addressed as talker and listener to "unaddress themselves." Once the UNTALK and UNLISTEN bytes have been sent, the 4027A turns off the ATN signal.

GPIB Code Chart

During ordinary data transfers over the GPIB, the 8-bit bytes are usually ASCII characters. When the 4027A is asserting ATN, however, the bytes transferred take on special meanings: they may be special commands (such as SPE, "serial poll enable"), or device addresses (such as LA1, "primary listen address for device number 1"). These alternate meanings are listed in the GPIB Code Chart, Table 4-12.

Table 4-12

GPIB CODE CHART

DIO7 DATA BUS DIO6 DIO5 DIO4 DIO3 BITS DIO2 DIO1				0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
				ADDRESSED COMMANDS	UNIVERSAL COMMANDS	PRIMARY LISTEN ADDRESSES		PRIMARY TALK ADDRESSES		SECONDARY ADDRESSES	
0	0	0	0	NUL 0	DCE 16	SP LA0 32	0 LA16 48	@ TA0 64	P TA16 80	' SA0 96	p SA16 112
0	0	0	1	SOH GTL 1	DC1 LLO 17	0 LA1 33	1 LA17 49	A TA1 65	Q TA17 81	a SA1 97	q SA17 113
0	0	1	0	STX 2	DC2 18	" LA2 34	2 LA18 50	B TA2 66	R TA18 82	b SA2 98	r SA18 114
0	0	1	1	ETX 3	DC3 19	# LA3 35	3 LA19 51	C TA3 67	S TA19 83	c SA3 99	s SA19 115
0	1	0	0	EOT SDC 4	DC4 DCL 20	$ LA4 36	4 LA20 52	D TA4 68	T TA20 84	d SA4 100	t SA20 116
0	1	0	1	ENQ PPC 5	NAK PPU 21	% LA5 37	5 LA21 53	E TA5 69	U TA21 85	e SA5 101	u SA21 117
0	1	1	0	ACK 6	SYN 22	& LA6 38	6 LA22 54	F TA6 70	V TA22 86	f SA6 102	v SA22 118
0	1	1	1	BEL 7	ETB 23	' LA7 39	7 LA23 55	G TA7 71	W TA23 87	g SA7 103	w SA23 119
1	0	0	0	BS GET 8	CAN SPE 24	(LA8 40	8 LA24 56	H TA8 72	X TA24 88	h SA8 104	x SA24 120
1	0	0	1	HT TCT 9	EM SPD 25	) LA9 41	9 LA25 57	I TA9 73	Y TA25 89	i SA9 105	y SA25 121
1	0	1	0	LF 10	SUB 26	* LA10 42	: LA26 58	J TA10 74	Z TA26 90	j SA10 106	z SA26 122
1	0	1	1	VT 11	ESC 27	+ LA11 43	; LA27 59	K TA11 75	[TA27 91	k SA11	{ SA27 123
1	1	0	0	FF 12	FS 28	, LA12 44	< LA28 60	L TA12 76	\ TA28 92	l SA12 108	! SA28 124
1	1	0	1	CR 13	GS 29	- LA13 45	= LA29 61	M TA13 77	] TA29 93	m SA13 109	} SA29 125
1	1	1	0	SO 14	RS 30	. LA14 46	> LA30 62	N TA14 78	^ TA30 94	n SA14 110	~ SA30 126
1	1	1	1	Si 15	US 31	/ LA15 47	? UNL 63	O TA15 79	_ UNT 95	o SA15 111	DEL 127

- Shaded codes are those usable by the 4027 with Options 4 and 36.

- LAn Primary Listen Address for device n
- TAn Primary Talk Address for device n
- UNL UNLISTEN command
- UNT UNTALK command
- SPE SERIAL POLL ENABLE command
- SPD SERIAL POLL DISABLE command

- KEY

CAN
SPE
24

ASCII Character
GPIB Code
Decimal

(2830) 2656-241

GPIB Firmware

The firmware controls the GPIB lines by reading and writing from the GPIB Board I/O registers. To illustrate the process, we'll consider three primitive operations: controlling and monitoring the GPIB lines, sending a data byte, and receiving a data byte. (More complicated procedures are made up of many of these primitive operations.)

Controlling and Monitoring the GPIB Lines

To control many of the GPIB signal lines, the firmware writes into the GPIB Control Word, X'0846'. For instance, to send the ATN signal, it writes a 1 into bit 3 of this word; to turn off ATN, it writes a 0 there.

Control Word Bits 1 through 5 control the five management bus lines. Bit 0 drives the HOLD signal, which is sent when it is necessary to delay completion of a transfer bus handshake cycle. (HOLD forces NRFD true.) Bit 6 steers the Handshake Logic (described later in this section) between "transmit" (source handshake) and "receive" (acceptor handshake) modes. Bit 7 controls the MYGPIBADR signal, by which the firmware tells the hardware that it has been addressed as a listener or talker for an upcoming GPIB data transfer.

To monitor the state of the GPIB lines, the firmware reads repeatedly from the GPIB Control Word. Bits 1 to 5 of this word tell the states of the five management bus lines. Bit 0 tells the state of the DAV line on the transfer bus. Bit 6 provides a warning flag (the NOBODY signal) whenever there are no addressed listeners attached to the bus. Bit 7 (the HAND signal) goes true when action is required of the Processor in order to complete a transfer bus handshake.

NOTE

Circuitry is provided to generate Processor interrupts whenever the states of signal lines change (Line Status Change Interrupt), or whenever Processor action is needed to complete a transfer bus handshake (HAND Interrupt). However, this circuitry is unused; the Option 36 firmware disables GPIB Board interrupts and monitors the signal lines by repeatedly reading from the GPIB Control Word.

Sending a Data Byte

To send a data byte over the GPIB, the firmware does the following:

1. First, it writes into the GPIB Control Word, setting the XMIT bit true. This steers the handshake Logic into its "transmit" mode.

2. Next, the firmware reads from the GPIB Control Word to be sure that the NOBODY signal is not being sent. (If NOBODY is true, then there are no listeners present on the GPIB, and the firmware aborts the attempt to send a data byte.)

3. Next, it checks the state of the HAND bit in the GPIB Control Word. HAND goes true when the GPIB "listeners" are ready for the 4027A to place a byte on the data bus.

4. When HAND goes true, the firmware writes a data byte into the GPIB Data Word. This places the data byte on the GPIB data bus.

5. After waiting for the data to settle, the firmware writes to the Trigger Word. This sends a SHAKE signal to the hardware. The hardware then completes the transfer bus handshake cycle. (When all the GPIB listeners have accepted the byte, HAND goes true again.)

6. The firmware repeats Steps 2 to 5 until all the data bytes have been sent.

Receiving a Data Byte

To receive a GPIB data byte, the firmware does the following:

1. First, it writes into the GPIB Control Word, setting the XMIT bit false. This steers the Handshake Logic into "receive" mode.

2. The firmware monitors the HAND bit in the GPIB Control Word. (The firmware repeatedly reads from this word to check its status.)

3. When the HAND bit goes true, the firmware reads the data byte from the GPIB Data Word, and places that data byte in the appropriate input queue.

4. Next, it writes to the Trigger Word, sending the SHAKE signal. This causes the GPIB handshaking to continue. When the next byte from the talker is ready in the GPIB Data Word, the Handshake Circuitry will set the HAND bit true again.

5. Steps 2 through 4 are repeated for successive data bytes.

Circuit Descriptions

Refer to Figure 6-13, the GPIB Interface Board block diagram. The circuitry includes these circuit blocks: Bus Data Buffer, Address Decoder, GPIB Bus Transceivers, GPIB Data Buffer and Latch, GPIB Control Word (Write), Debouncers, HAND Decode, Hello Gate, GPIB Control Word (Read), ATN Mask, Handshake Steering, and Handshake Logic.

Also included are "interrupt control" circuit blocks: Line Status Change Detector, Interrupt Mask Logic, Interrupt Requestor, and the buffers which form the "read" half of the GPIB Board Status Word. However, since the Option 36 firmware disables GPIB Board interrupts, most of this circuitry is unused.

Bus Data Buffer

The Bus Data Buffer (Schematic 17-1) consists of bidirectional tri-state buffers which interface the Mother Board data bus (BD0-BD7) with the GPIB Board's internal data bus (D0-D7). These buffers are steered by the PROCREAD (Processor Read) signal from the Address Decoder.

Address Decoder

The Address Decoder (Schematic 17-1) monitors the BA0-BA9, IOARD, READ, and WRITE lines, and detects when the Processor is reading or writing to the various I/O registers on the GPIB Interface Board. Its outputs are:

- **DRSTROBE.** Indicates a read from the GPIB Data Word.
- **SHAKE.** Generated by a write to the Trigger Word; indicates that the Processor has performed the handshaking function requested by the HAND signal.
- **LSCCLR.** Generated by a read from the Trigger Word; causes the LSC (Line Status Change) interrupt bit to be cleared.
- **CWSTROBE.** Indicates a write to the GPIB Control Word.
- **CRSTROBE.** Indicates a read from the GPIB Control Word.

I/O Registers

The GPIB Interface has four I/O registers through which it communicates with the Processor. These registers and their functions are listed in Table 4-13.

Table 4-13

GPIB BOARD I/O REGISTERS

Register Name	Address	Bit	Read Function	Write Function
GPIB Board Status Word	X'0840'	0	LSC interrupts are enabled	Enable LSC interrupts
		1	An LSC interrupt has occurred	unused
		2	HAND interrupts are enabled	Enable HAND interrupts
		3	A HAND interrupt has occurred	unused
		4-6	unused	unused
		7	An interrupt has occurred	unused
Trigger Word	X'0842'	----	Clear any LSC interrupts	Send SHAKE to the Handshake Logic
GPIB Data Word	X'0844'	0-7	Data byte received from the GPIB	Data byte to send on the GPIB
GPIB Control Word	X'0846'	0	DAV (Data Valid)	HOLD: Prevents completion of transfer bus handshake
		1	EOI (End or Identify)	same as read
		2	SRQ (Service Request)	same as read
		3	ATN (Attention)	same as read
		4	IFC (Interface Clear)	same as read
		5	REN (Remote Enable)	same as read
		6	NOBODY: True when "nobody is listening" on the GPIB	XMIT: true=send data bytes, false=receive data bytes
		7	HAND: True when a data transfer is awaiting Processor action	MYGPIBADR (My GPIB Address) tells the interface it's been addressed as the talker or listener

GPIB Transceivers

The GPIB Transceivers (Schematic 17-1, Figure 4-35) are special integrated circuits which interface the TTL logic on the circuit board with the active-low GPIB signal lines. Each has four terminals (A, B, C, D) which connect to the GPIB; four "incoming data" terminals (AI, BI, CI, DI) which drive on-board versions of the GPIB signals; and four "outgoing data" terminals (A0, B0, C0, D0) which drive the GPIB lines, when enabled by a low at the "enable" pins.

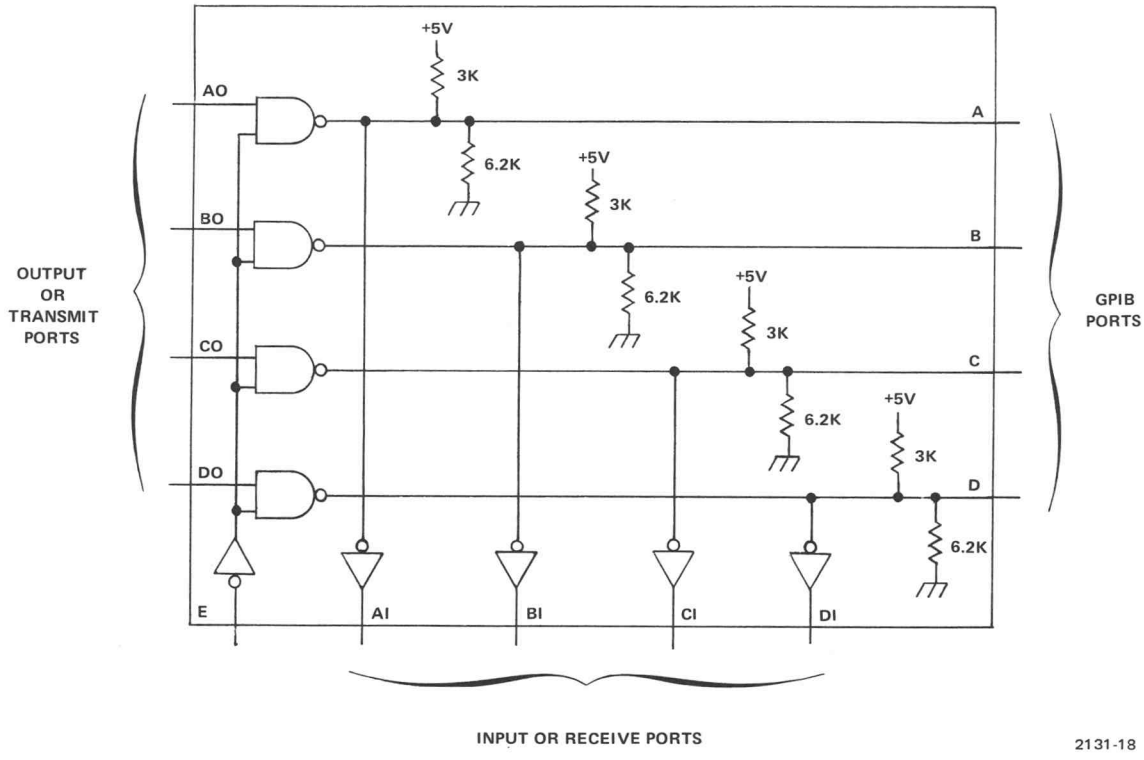


Figure 4-35. GPIB Transceiver.

GPIB Data Buffer and Latch

The GPIB Data Buffer and Latch (Schematic 17-1) includes the "read" and "write" halves of the GPIB Data Word, and GPIB transceiver ICs to interface them to the GPIB data bus.

The "read" half of the GPIB data word is an IC containing eight tri-state buffers. When the Processor reads from the GPIB Data Word, the DRSTROBE (Data Read Strobe) signal enables these buffers, placing the current GPIB data byte on the board's internal data bus. The Data Bus Buffers then relay the data to the BD0-BD7 Mother Board data lines.

The "write" half of the GPIB Data Word is an IC containing eight type D flip-flops. When the Processor writes to the Data Word, the DWSTROBE (Data Write Strobe) signal clocks these flip-flops, storing the data in them. The DXMIT signal (from the Handshake Steering logic) enables the GPIB Transceiver ICs, placing the data on the GPIB data bus.

"Write" Half of GPIB Control Word

The "write" half of the GPIB Control Word (Schematic 17-1) is an IC with eight Type D flip-flops. When clocked by CWSTROBE (Control Word Write Strobe), the flip-flops store the data on the board's internal data bus. The flip-flops drive these signal lines:

- **HOLD.** A signal to the Handshake Logic, causing it to force the NRFD (Not Ready For Data) signal true. This prevents completion of the current transfer bus handshake.
- **EOIRQST (EOI Request).** Requests the Handshake Logic to send the EOI (End or Identify) signal on the GPIB. (If ATN is true, the EOI signal is suppressed.)
- **SENDSRQ, SENDATN, SENDIFC, SENDREN.** These signals directly drive the GPIB Transceivers, sending the SRQ, ATN, IFC, and REN signals.
- **MYGPIBADR (My GPIB Address).** A signal to the handshake Logic that the 4027A is to take part as a talker or listener in data transfers.

Debouncers

Before incoming GPIB signals are passed on to the rest of the GPIB Board circuitry, they are "debounced" (removed of short-duration noise). The Debouncers (Schematic 17-2) for six of the signal lines use an MC14490 "hex contact bounce eliminator" IC; for the seventh signal line (IFC), a network of gates and smoothing capacitors is used. The debounced outputs are: DNDAC (Debounced Not Data Accepted), DEOI (Debounced End or Identify), DDAV (Debounced Data Valid), DATN (Debounced Attention), DSRQ (Debounced Service Request), DIFC (Debounced Interface Clear), and DHAND (Debounced HAND).

HAND Decode

The HAND Decode logic (Schematic 17-2) monitors two of the transfer bus handshake lines (NRFD and DAV).

When **sending data (DXMIT true)**, HAND goes true when NRFD (Not Ready For Data) goes false. In this case, HAND tells the Processor, "The listeners are ready for data; go ahead and give them another data byte."

When **receiving data (XMIT false)**, HAND goes true when DAV (Data Valid) goes true. In this case, HAND tells the Processor, "The talker has placed a valid data byte on the bus; go ahead and read the byte from the GPIB Data Word."

Hello Gate

The Hello Gate (Schematic 17-2) monitors the two transfer bus lines (NRFD and NDAC) which are driven by listeners in a GPIB data transfer. As long as there is a GPIB listener present, at least one of these signals will be true; but if neither of NRFD or NDAC is true, then "there's nobody out there listening," and the gate sends the NOBODY signal. The Processor reads the NOBODY bit (from the GPIB Control Word) before attempting to send data over the GPIB.

"Read" Half of GPIB Control Word

The "read" half of the GPIB Control Word (Schematic 17-2) consists of tri-state buffers. When enabled by the CRSTROBE (Control Word Read Strobe) signal, these buffers place bits on the data bus bits (a) tell the state of several GPIB signal lines (EOI, DAV, ATN, SRQ, IFC, REN signals), (b) warn when there are no listeners on the bus (NOBODY signal), and (c) tell the Processor when action is required of it to complete a GPIB handshake (HAND signal).

ATN Mask

The ATN Mask gate (Schematic 17-1) monitors the GPIB ATN signal and the SENDATN from the GPIB Control Word. If ATN is true, and it is **not** the 4027A which is sending it true (SENDATN is false), the gate sends HESEZATN (He Says Attention). This signal is used by the Handshake Steering logic.

NOTE

The Option 36 firmware requires that the 4027A be the only controller (and thus the only device allowed to send ATN) on the GPIB. Thus, HESEZATN should always be false.

Handshake Steering

The Handshake Steering logic (Schematic 17-1) enables the "source handshake" part of the Handshake Logic (by sending the DXMIT signal) if the firmware has asked to transmit data (XMIT true) and there is no other GPIB device sending the ATN signal (HESEZATN is false).

The logic enables the "acceptor handshake" part of the Handshake Logic (by sending LISTEN true) if an external controller sends ATN (HESEZATN is true) or if the 4027A is not sending data (XMIT is false).

NOTE

Since the 4027A is the only controller allowed on the GPIB, HESEZATN should always be false.

Handshake Logic

The Handshake Logic (Schematic 17-1) includes circuitry for performing the GPIB "source handshake" function when sending data bytes, and for performing the "acceptor handshake" function when receiving data bytes.

Source Handshake. The 4027A sends data bytes over the GPIB as follows:

1. When all listeners are ready for a data byte, the GPIB NRFD (Not Ready For Data) signal goes false. This is detected by the HAND Decode logic, which sets the HAND bit in the GPIB Control Word, inviting the Processor to place the data byte on the GPIB.

2. Meanwhile, the Processor has been reading repeatedly from the Control Word. When the HAND bit goes true, it writes into the GPIB Data Word, placing a data byte on GPIB data lines DIO1-DIO8.

3. After the data has settled, the Processor writes to the Trigger Word. This causes the Address Decoder to send the SHAKE signal to the Handshake Logic.

4. SHAKE sets the source handshake flip-flop (U270A in Schematic 17-1). With the Handshake Logic steered to "source handshake" mode (DXMIT true), the flip-flop output passes through an AND gate, which sends SENDDAV to the GPIB Bus Transceivers. This causes the 4027A to send the DV (Data Valid) signal on the GPIB.

5. The GPIB listeners set NRFD true again and read the data byte.

6. When the slowest listener has accepted the data, NDAC (Not Data Accepted) goes false. The debounced version of this signal (DNDAC) clears the source handshake flip-flop, causing the 4027A to stop sending DAV.

7. When all listeners are ready for the next data byte, NRFD goes false again. Steps 1 through 6 then repeat for successive data bytes.

Acceptor Handshake. The 4027A receives data bytes from the GPIB as follows:

1. When the talker has placed a byte on the GPIB data bus (DIO1-DIO8), it sends the DAV (Data Valid) signal. With the GPIB board steered to "receive" mode (XMIT false), the HAND Decode logic sets the HAND bit in the GPIB Control Word.

   Several logic gates (U190D, U180A, U290A, U290D) send SENDNDAC as long as either DAV or HAND is true. This causes the 4027A to send the NDAC (Not Data Accepted) signal on the GPIB.

2. Meanwhile, the Processor has been reading repeatedly from the GPIB Control Word. When HAND goes true, it reads from the GPIB Data Word.

3. Having read the data, the Processor writes to the Trigger Word. This causes the Address Decoder to send the SHAKE signal to the Handshake Logic.

4. SHAKE sets the acceptor handshake flip-flop (U270B in Schematic 17-1). As the flip-flop is set, its inverting output goes false; this turns off the SENDNDAC signal. This causes the 4027A to stop sending NDAC (Not Data Accepted).

5. When all listeners have accepted the data, the GPIB NDAC line goes false (high). This tells the talker that the listeners have the data. The talker then turns off DAV.

6. When DAV goes false, the HAND Decode logic stops sending the HAND signal. With both HAND and DAV false, logic gates in the Handshake Logic stop sending the SENDNRFD signal (unless the HOLD bit in the GPI Control Word is set).

   Also, as DAV goes false, its debounced version (DDAV) clears the acceptor handshake flip-flop. With the flip-flop cleared, SENDNDAC goes true, and the 4027A sends the NDAC (Not Data Accepted) signal on the GPIB.

7. When all listeners have stopped sending NRFD, the talker places another data byte on the GPIB and sends DAV true again. Steps 1 through 6 then repeat for successive data bytes.

Interrupt Control Circuitry

The GPIB Board's interrupt control circuitry includes: Line Status Change Detector, Interrupt Mask Logic, Interrupt Requestor, and the "read" half of the GPIB Board Status Word. This circuitry provides for two types of Processor Interrupts (Line Status Change Interrupts and HAND Interrupts).

NOTE

The Option 36 firmware disables these interrupts. Therefore, most of this circuitry is unused.

LSC Detector. The LSC Detector (Schematic 17-2) is unused.

Interrupt Mask Logic. The Interrupt Mask Logic (Schematic 17-2) includes interrupt latches (flip-flops U360A and U360B) and latches for the interrupt enable bits (U220A and U220B). The firmware disables interrupts by writing zeroes into bits 0 and 2 of the GPIB Status Word. This causes the Address Decoder to send SWSTROBE (Status Word Strobe), loading the zeroes (bits 0 and 2 of the word written) into the interrupt enable latches. This sets the HINTEN (HAND Interrupt Enable) and LSCINTEN (Line Status Change Interrupt Enable) bits false. With HINTEN and LSCINTEN false, the HAND and LSC interrupts are both disabled.

"Read" Half of Board Status Word. The "read" half of the Board Status Word (Schematic 17-2) is unused.

Interrupt Address Decoder. With interrupts disabled, the Interrupt Address Decoder is unused.

OPTION 22, DISPLAY MEMORY EXPANSION

Option 22 provides 32K bytes of Display Memory.

The standard complement of RAM on the Display Memory Board is 16K.

NOTE

Installation information for Option 22 can be found in Volume 2 of this Service Manual.

OPTIONS 27, 28, AND 29, GRAPHICS MEMORY EXPANSION

Options 27, 28 and 29 provide 32K words, 48K words, and 64K words of Graphics Memory, respectively.

It is possible to install up to four 16K-by-24 bit banks of RAM on the Graphics Memory Board. The standard complement of RAM is 16K; installed at the highest addresses in character memory address space (Bank 3). Memory expansion takes place from the top downward, in 16K increments.

NOTE

Installation information for Options 27, 28, and 29 can be found in Volume 2 of this Service Manual.

OPTION 31, CHARACTER SET EXPANSION BOARD

The Character Set Expansion board allows up to eight alternate ROM-contained character sets to be installed in the 4027A. Each of these character sets occupies a "font" in character memory address space. The Character Set Expansion board has space for up to eight fonts of 128 characters each. (A ROM font is 2K, eight bit bytes long.)

All addressing and control functions for the Character Set Expansion board are performed by the Graphics Memory Controller board; the logical interface is by means of a 34-conductor cable to that board. Only power is derived from the Mother Board.

ROM Memory Cycle

Figure 4-36 depicts the waveforms generated on the Graphics Memory Controller board for a typical ROM memory cycle. There are eight font set select signals (CS0 thru CS7) each of which selects a particular font (either 1 or 2 ROMs) on the Character Set Expansion board. In addition, there are eleven address lines (ROM0 through ROM10) which select locations within the ROMs.

The +5V power to the ROMs is switched off and on by 6600/6601 power switching devices. When one of the select signals CS0 through CS7 goes true, two 6600/6601s turn on the power to the ROMs associated with that font.

Addressing Modes

There are two addressing modes for each font: modes 1 and 2. Mode 1 is used when a character set is contained on one ROM (64 characters or less) and mode 2 is used when the character set is contained on two ROMs (up to 128 characters). In mode 1, the character data is repeated in the upper and lower halves of the font. In mode 2, the font is addressed as 128 distinct character locations with different character data coming from each location.

Figure 4-37 shows a simplified schematic of the mode select circuitry for font 1 (see also the Schematics 21-1 and 21-2.) Notice that in **mode 2**, address line ROM10-0, after passing through the etched circuit bridge in J22, enables **ROM 1** for all addresses in the **upper** half of the font. Its complement, ROM10-1 enables **ROM 2** for all addresses in the **lower** half of the font. Thus the two ROMs are addressed sequentially, providing 2048 different locations.

For **mode 1** operation there is a jumper on J22 connecting the CS1-0 line to the ROM10-0 line. The effect of this is to **force ROM 1** to remain enabled for all addresses in the **upper and lower** halves of the font. In mode 1, only one ROM is installed; this in the socket corresponding to the upper half of the font.

NOTE

There is additional information on Option 31 ROM installation in Volume 2 of this Service Manual.

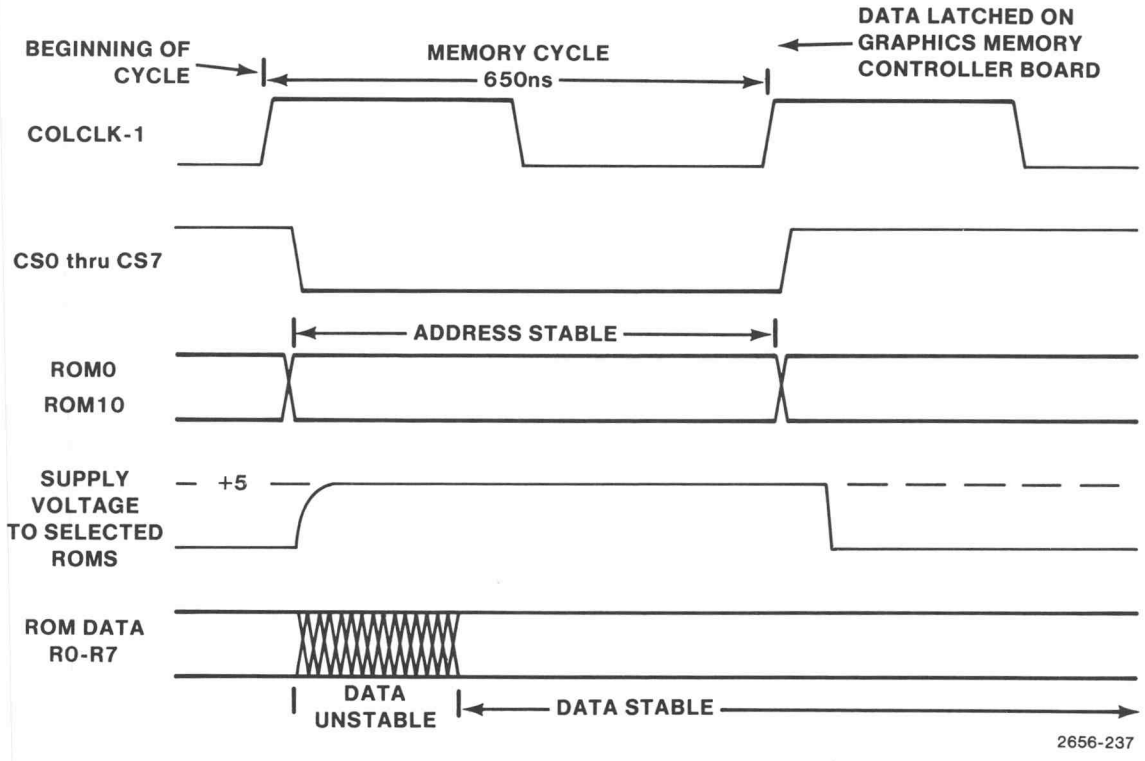


Figure 4-36. ROM Memory Cycle.

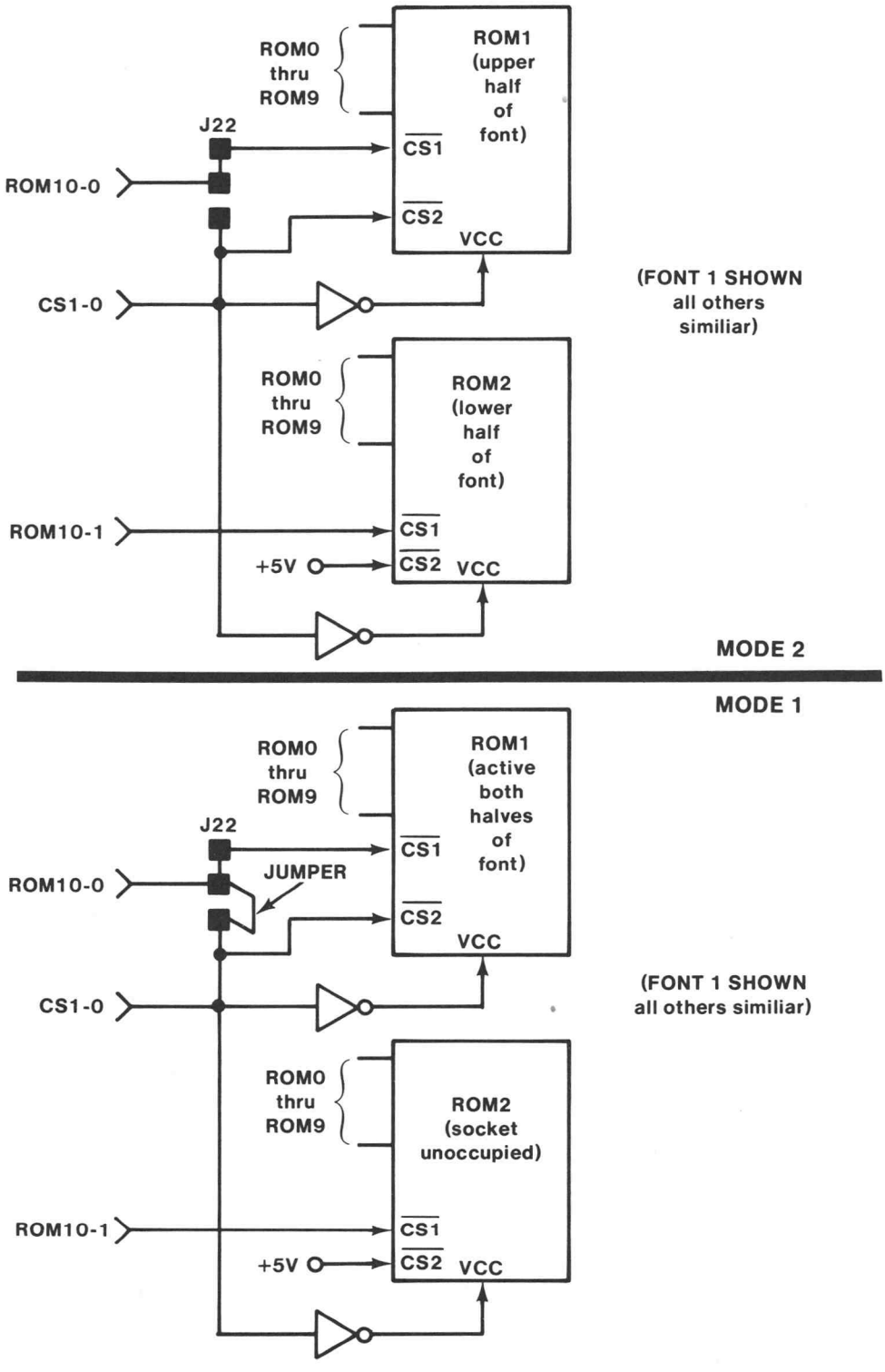


Figure 4-37. Character Set Expansion Board Modes 1 and 2.

OPTION 32, RULINGS CHARACTER SET

NOTE

Refer to Volume 2 of this Service Manual for Option 32 installation information.

Option 32 consists of a ROM holding bit patterns for rulings characters. The rulings character set typically occupies font 1 in the character memory address space. Figure 4-38 shows the characters contained in Option 32.

OPTION 34, MATH CHARACTER SET

NOTE

For Option 34 installation information see Volume 2 of this Service Manual.

Option 34 consists of a ROM holding bit patterns for the math character font. The math character font typically occupies font 3 in character memory address space. Figure 4-39 shows the characters contained in Option 34.

OPTION 36, PERIPHERALS ROM

Option 36 consists of two ROMs which hold the firmware for use with the RS-232 and GPIB Peripheral Interfaces (Options 3 and 4). These ROMs are installed on the Processor Board.

NOTE

Information on the installation of Option 36 is contained in Volume 2 of this Service Manual.

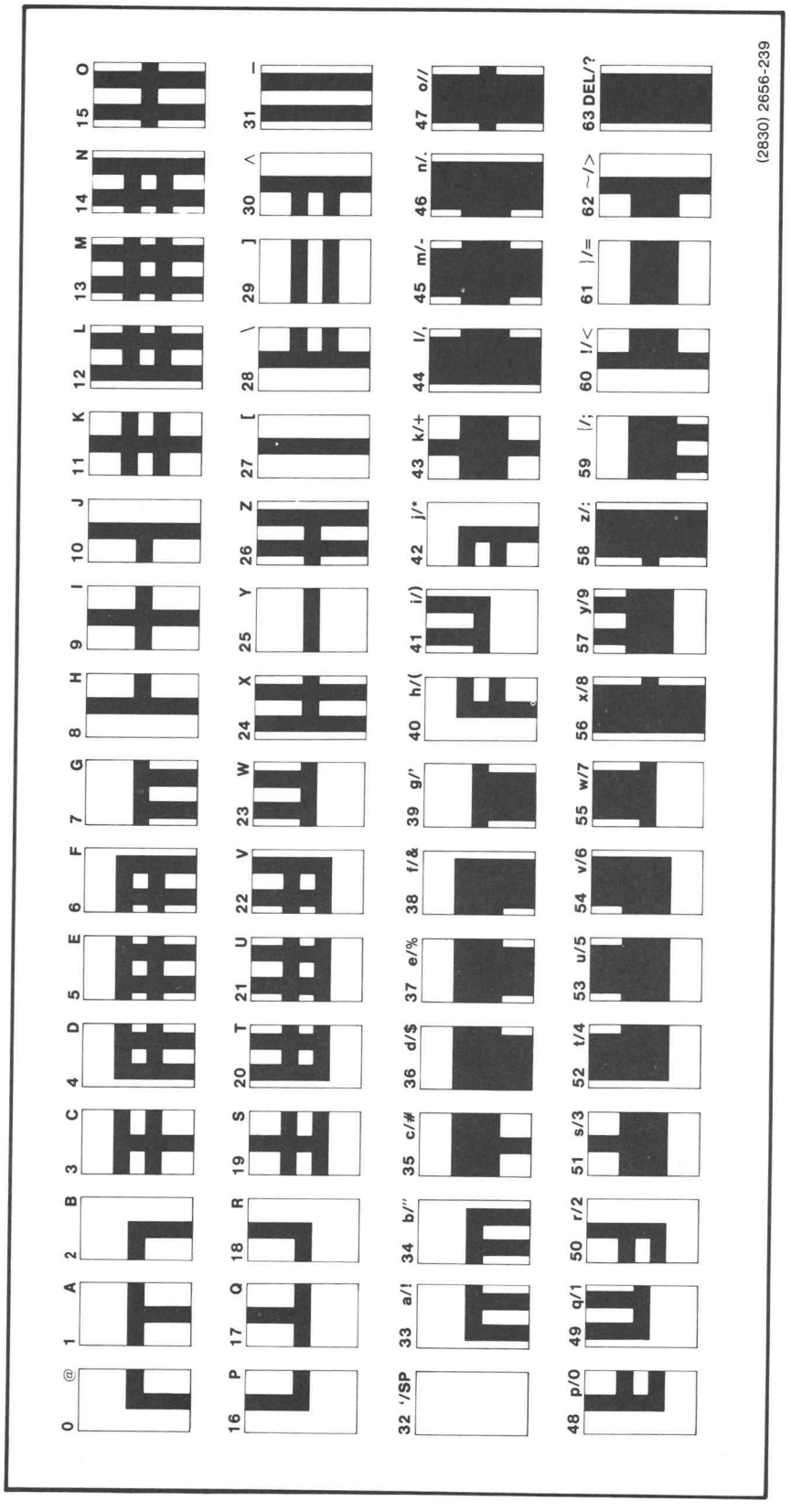


Figure 4-38. Rulings Characters.

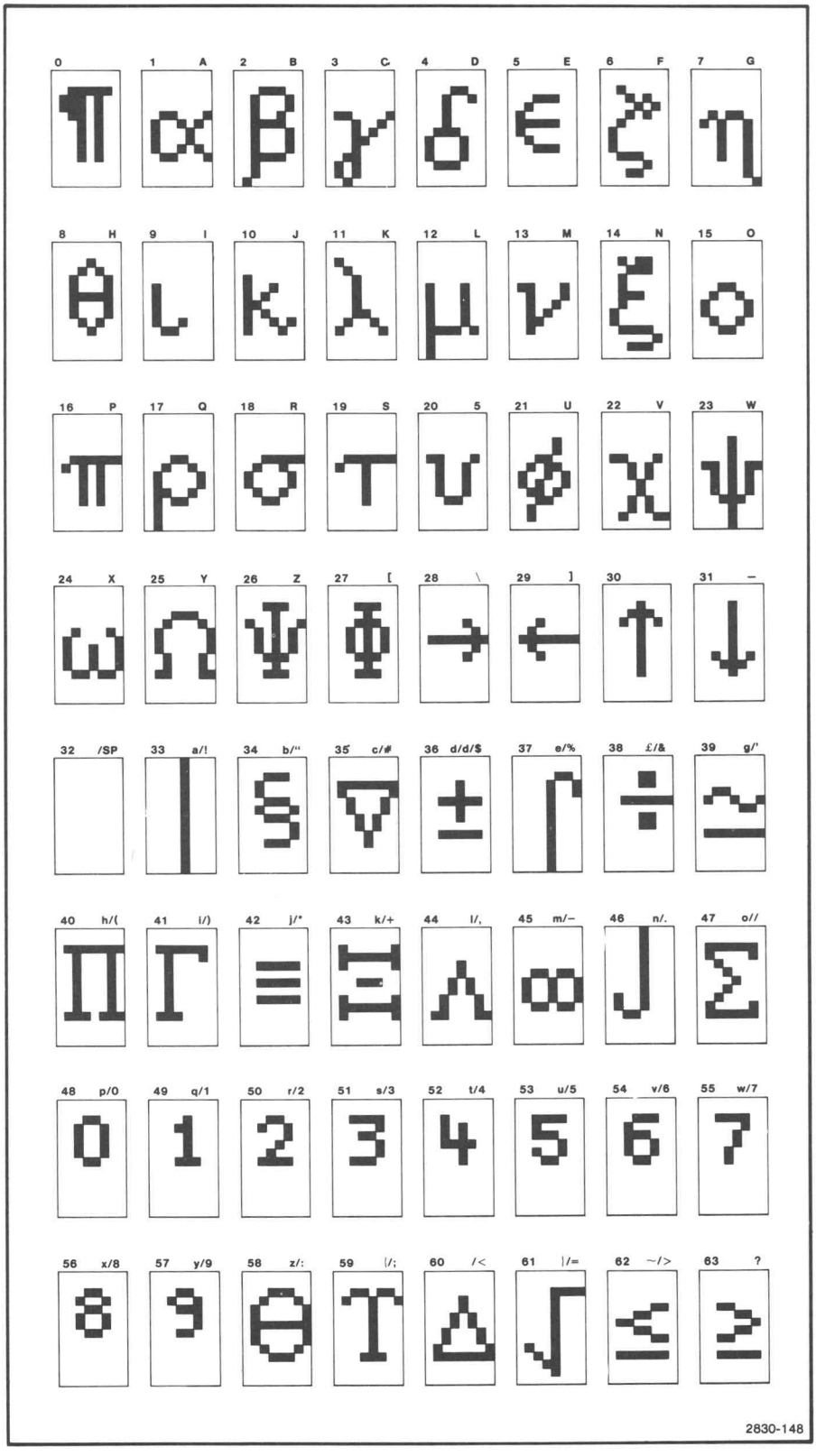


Figure 4-39. Math Characters.

HARD COPY AND VIDEO OUTPUT

Hard copy and video output consists of a set of cables and connectors which makes several video and synchronizing signals generated on the Display Controller Board available at the rear panel of the terminal.

A multipin connector (J5200) provides video and logic signals for driving a TEKTRONIX 4612 or 4632 Video Hard Copy Unit, and there are six BNC connectors (J5300 through J5800) which supply composite video, blanking, and sync signals for use by external video systems. For a discussion of the circuits involved in generating these signals, refer to the Display Controller circuit descriptions. Figure 4-40 depicts the cables and connectors and lists the signals they carry. Volume 2 of this service manual contains additional information.

Option	Description
A1	220V/16A 50Hz Operation Universal European plug.
A2	240V/13A 50Hz Operation United Kingdom plug.
A3	240V/10A 50Hz Operation Australian plug.
A4	240V/15A 60Hz Operation North American plug.

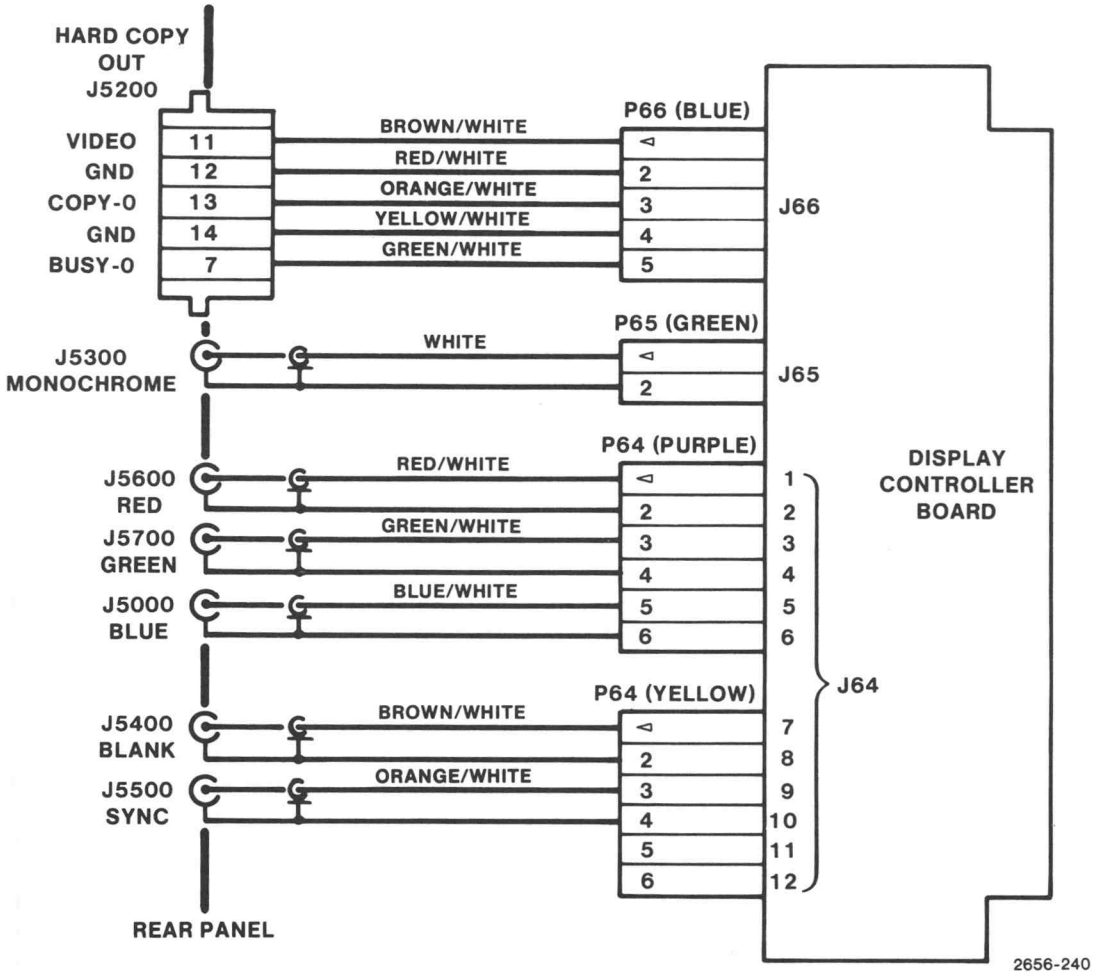


Figure 4-40. Rear Panel Connectors.

OPTION 4A, UNITED KINGDOM CHARACTER SET

This option permits TEKTRONIX 4020 Series terminals to change to the United Kingdom standard keyboard layout. The only change is that the "#" sign is replaced by the English " £ " sign. When this key is pressed (or the appropriate code is received by the terminal) the " £ " sign is displayed on the screen. The ASCII code for the " £ " sign is the same as that for the "#" sign.

OPTION 4B, FRENCH CHARACTER SET

This option changes TEKTRONIX 4020 Series terminals to the French "AZERTY" keyboard layout. All the characters are the same as on the standard keyboard. The only change is that four keys are switched around. See Figure 4-41. There are no changes to the ASCII Code Chart.

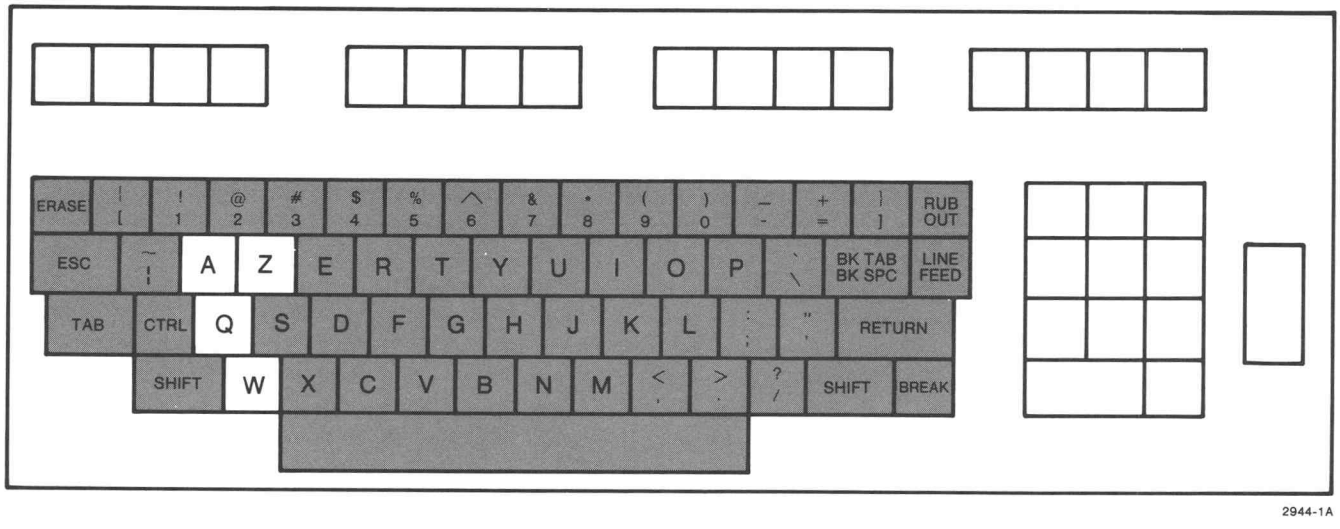


Figure 4-41. French Keyboard.

OPTION 4C, SWEDISH CHARACTER SET

This option changes TEKTRONIX 4020 Series terminal keyboards to the Swedish Standard layout and allows the Swedish characters to be displayed. There are seventeen changes to the keyboard; three of these changes are new characters. The changes are shown in the revised keyboard configuration (Figure 4-42), and the revised ASCII Code Chart (Table 4-14). When these seventeen keys are pressed (or the appropriate codes are received by the terminal), the corresponding characters are displayed on the screen.

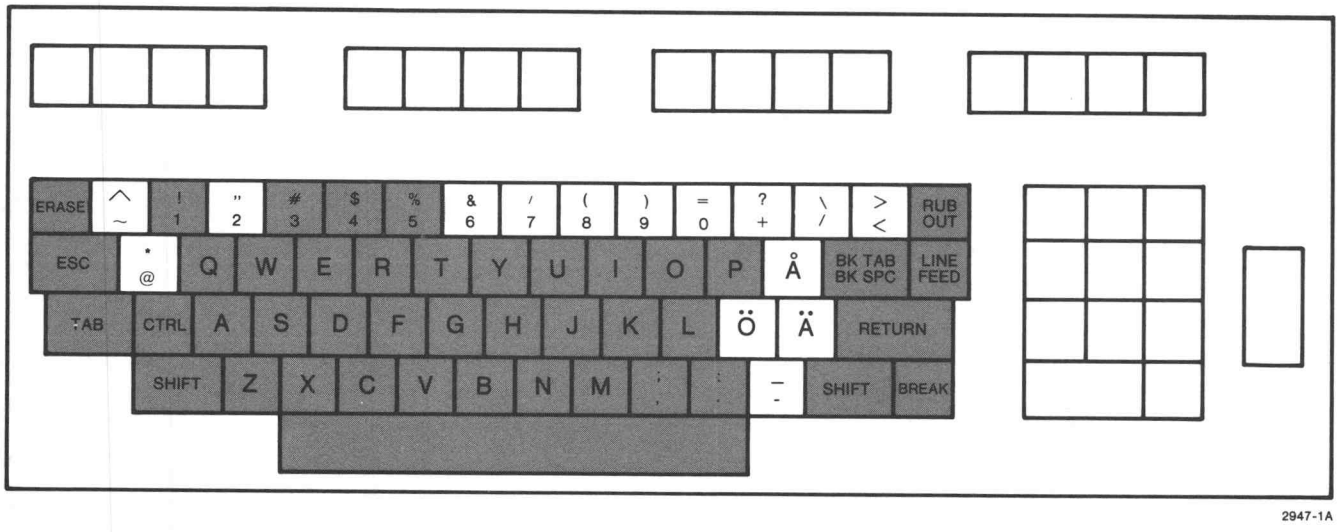


Figure 4-42. Swedish Keyboard.

Table 4-14

SWEDISH CHARACTER SET

B7 B6 B5 / BITS B4 B3 B2 B1	Ø Ø Ø	Ø Ø 1	Ø 1 Ø	Ø 1 1	1 Ø Ø	1 Ø 1	1 1 Ø	1 1 1
	CONTROL		HIGH X & Y GRAPHIC INPUT		LOW X		LOW Y	
Ø Ø Ø Ø	NUL 0	DLE 16	SP 32	0 48	@ 64	P 80	\ 96	p 112
Ø Ø Ø 1	SOH 1	DC1 17	! 33	1 49	A 65	Q 81	a 97	q 113
Ø Ø 1 Ø	STX 2	DC2 18	" 34	2 50	B 66	R 82	b 98	r 114
Ø Ø 1 1	ETX 3	DC3 19	# 35	3 51	C 67	S 83	c 99	s 115
Ø 1 Ø Ø	EOT 4	DC4 20	$ 36	4 52	D 68	T 84	d 100	t 116
Ø 1 Ø 1	ENQ 5	NAK 21	% 37	5 53	E 69	U 85	e 101	u 117
Ø 1 1 Ø	ACK 6	SYN 22	& 38	6 54	F 70	V 86	f 102	v 118
Ø 1 1 1	BEL 7	ETB 23	' 39	7 55	G 71	W 87	g 103	w 119
1 Ø Ø Ø	BS 8	CAN 24	(40	8 56	H 72	X 88	h 104	x 120
1 Ø Ø 1	HT 9	EM 25	) 41	9 57	I 73	Y 89	i 105	y 121
1 Ø 1 Ø	LF 10	SUB 26	* 42	: 58	J 74	Z 90	j 106	z 122
1 Ø 1 1	VT 11	ESC 27	+ 43	; 59	K 75	Ä 91	k 107	ä 123
1 1 Ø Ø	FF 12	FS 28	, 44	< 60	L 76	Ö 92	l 108	ö 124
1 1 Ø 1	CR 13	GS 29	- 45	= 61	M 77	Å 93	m 109	å 125
1 1 1 Ø	SO 14	RS 30	. 46	> 62	N 78	^ 94	n 110	~ 126
1 1 1 1	SI 15	US 31	/ 47	? 63	O 79	_ 95	o 111	RUBOUT (DEL) 127

2947-2A

Section 5
BLOCK DIAGRAMS

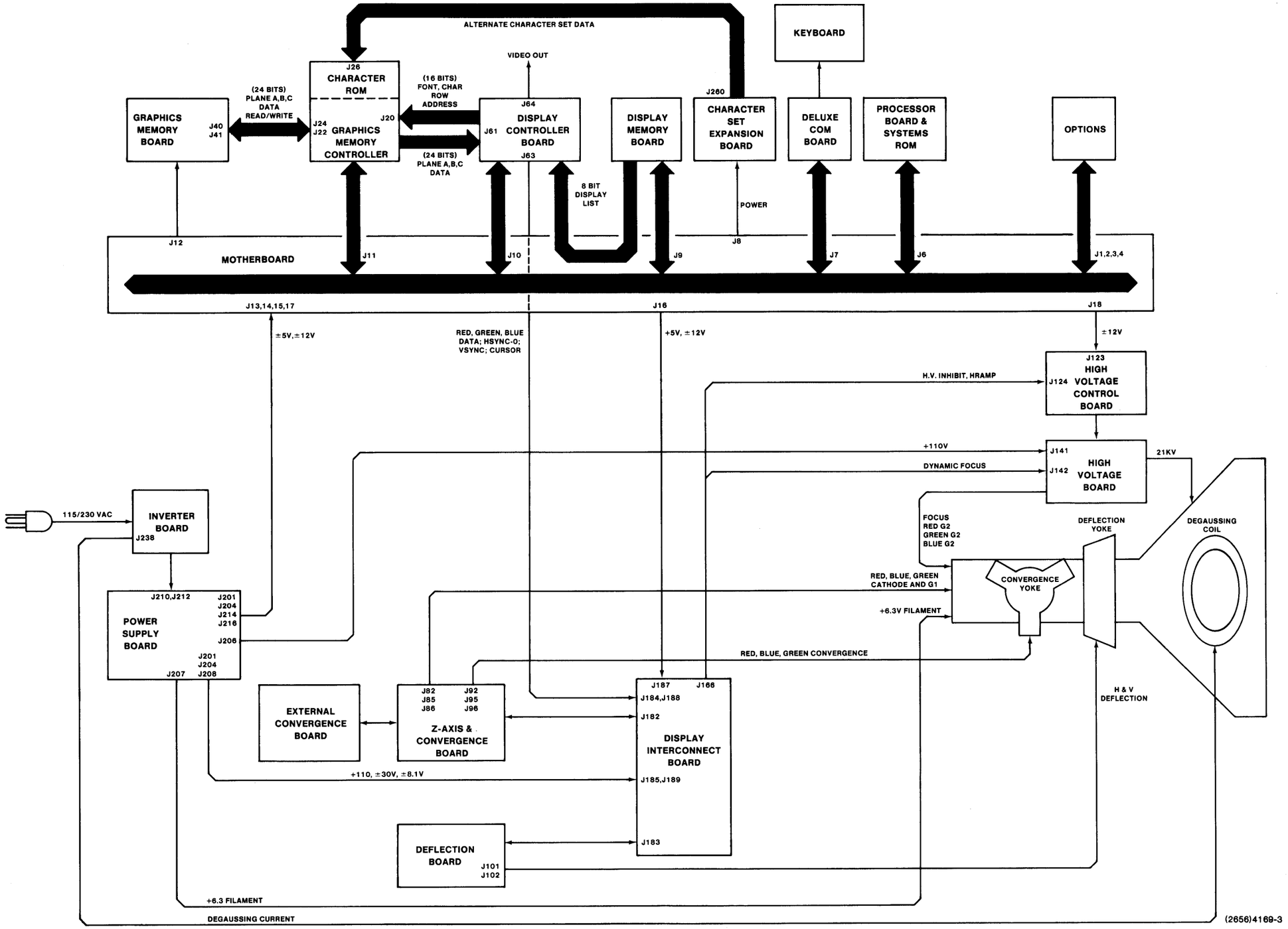


Figure 5-1. System Block Diagram.

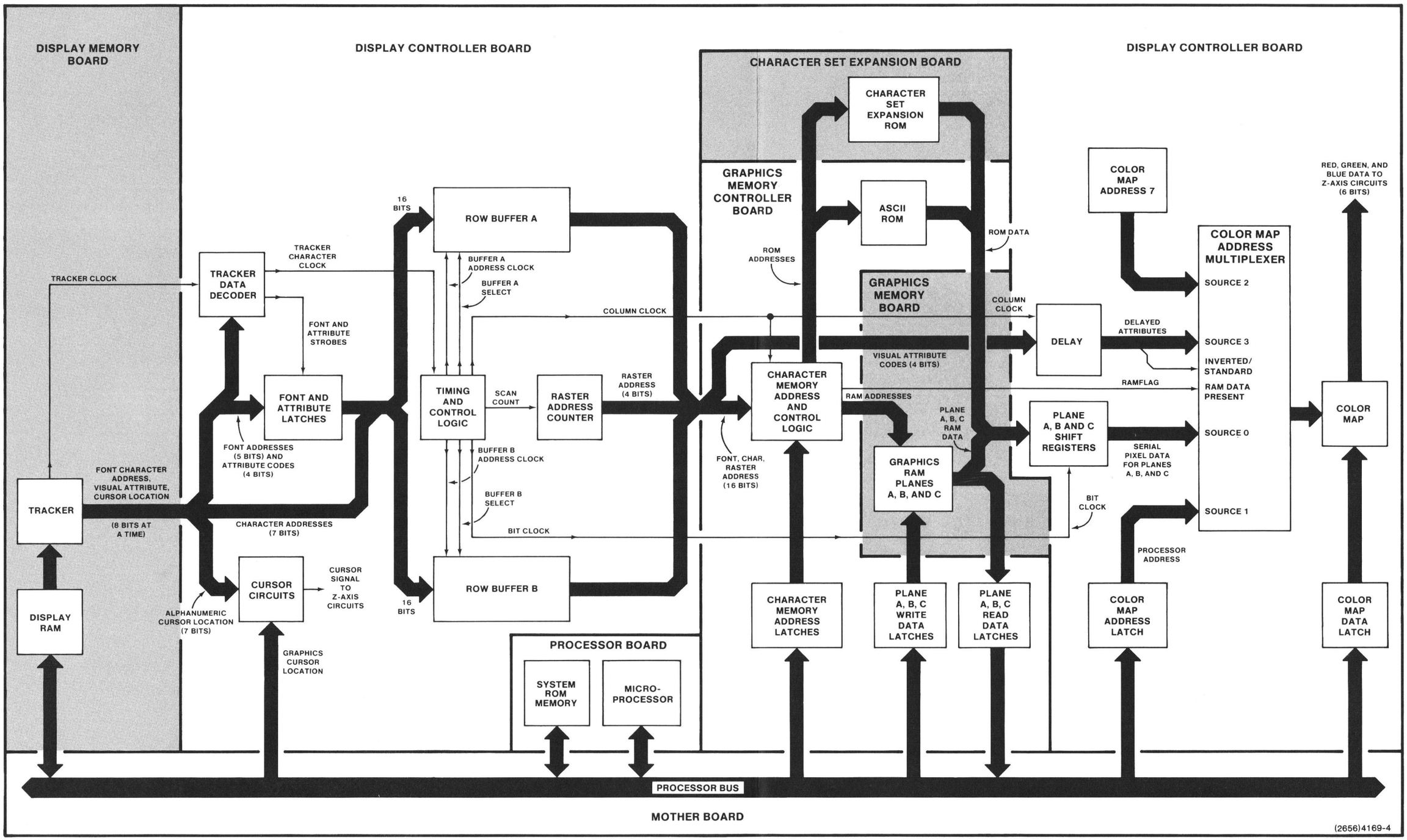
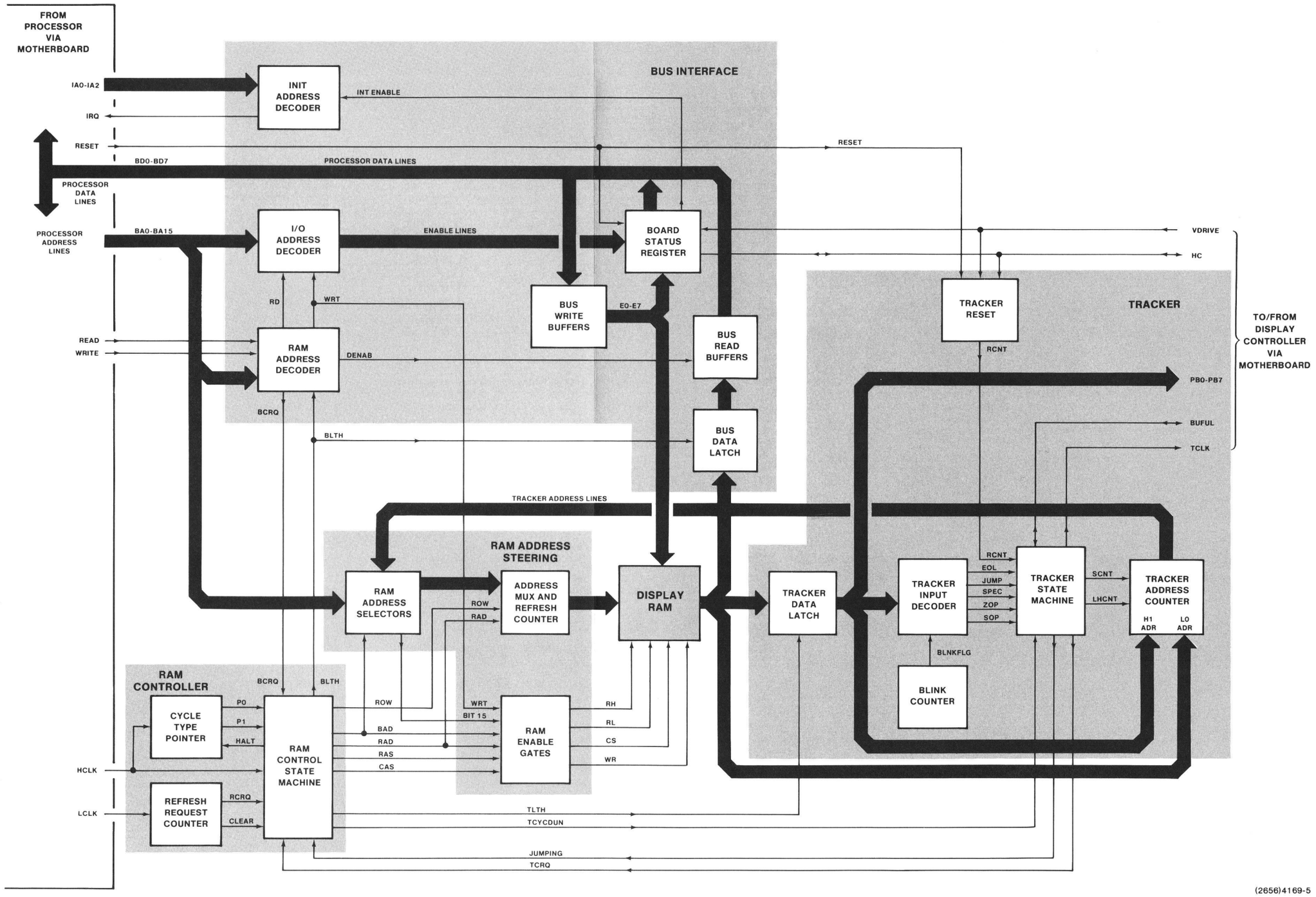


Figure 5-2. Display Processor Block Diagram.



(2656)4169-5

Figure 5-3. Display Memory Block Diagram.

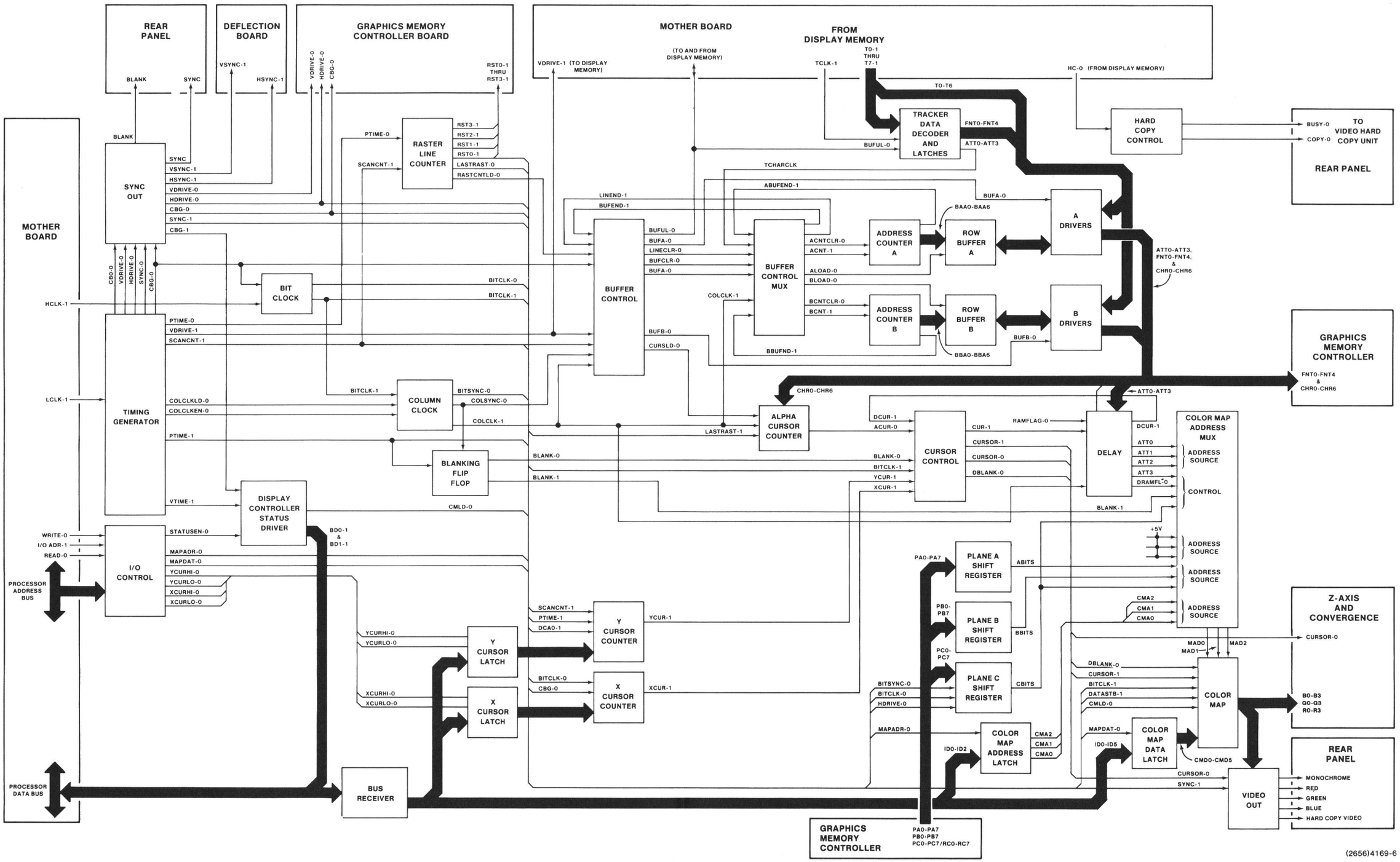


Figure 5-4. Display Controller Block Diagram.

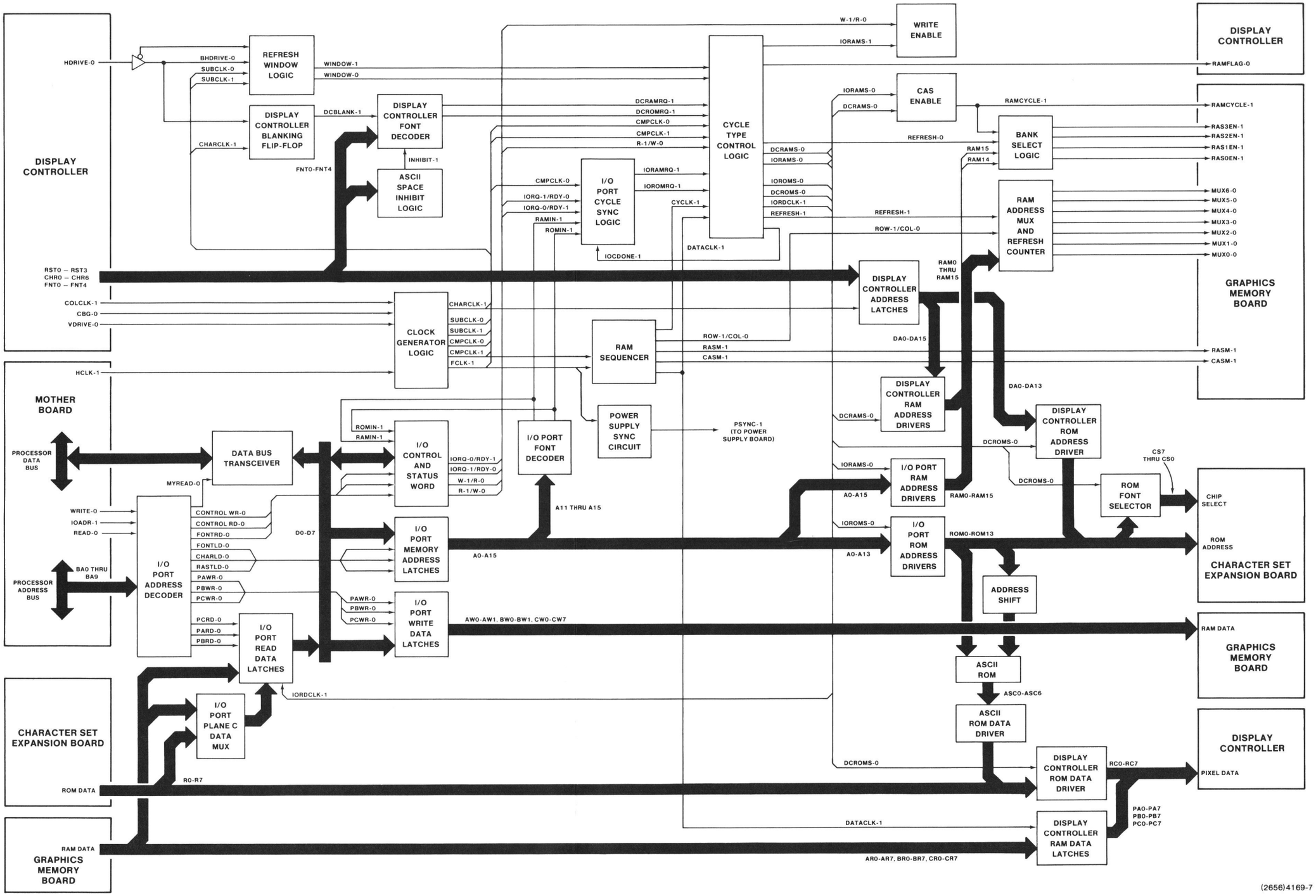


Figure 5-5. Graphics Memory Controller Block Diagram.

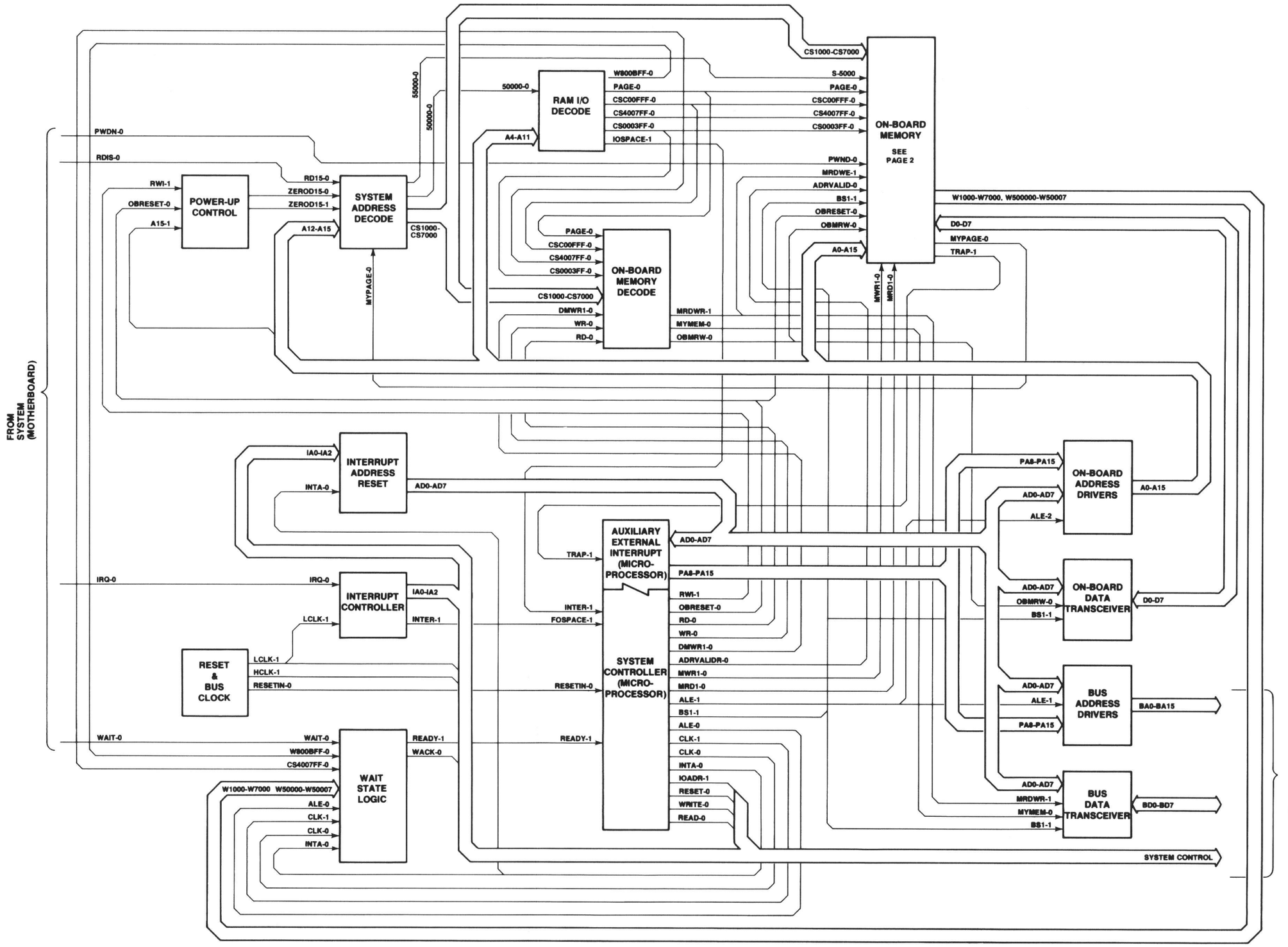


Figure 5-6A. Processor Block Diagram (Control and Address).

Figure 5-6A and 5-6B PROCESSOR BLOCK

4169-8

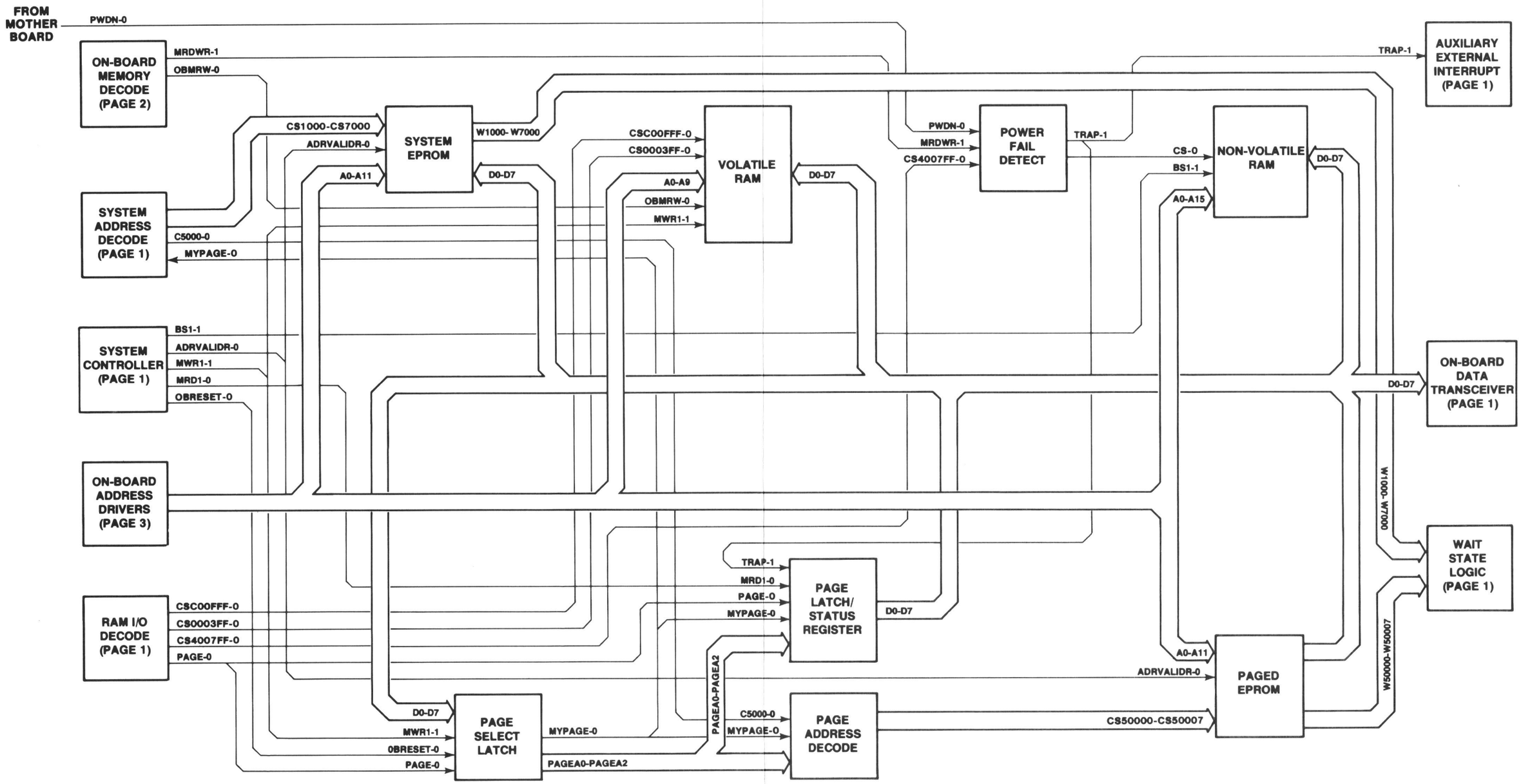


Figure 5-6B. Processor Block Diagram (Memory).

4169-9

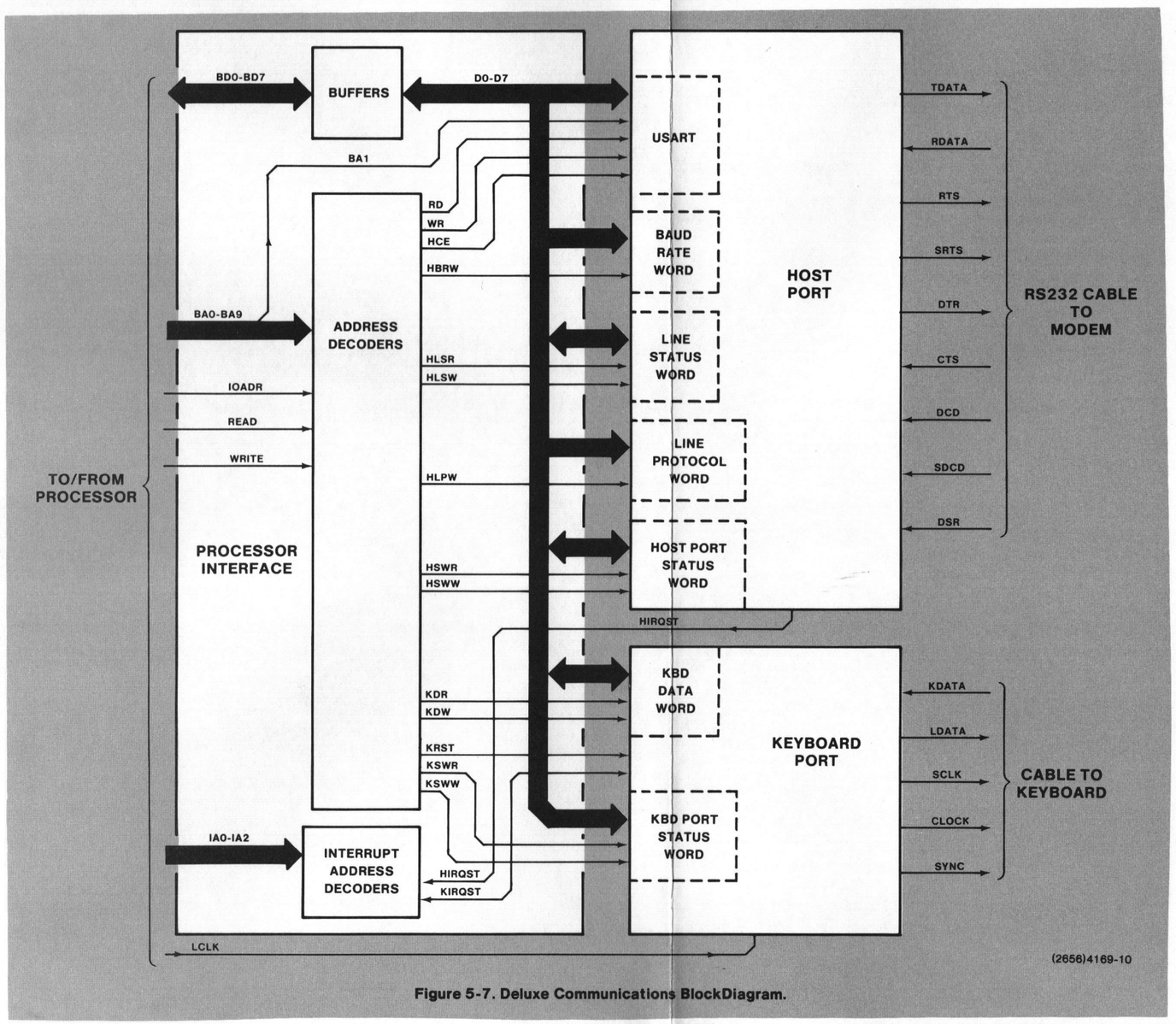
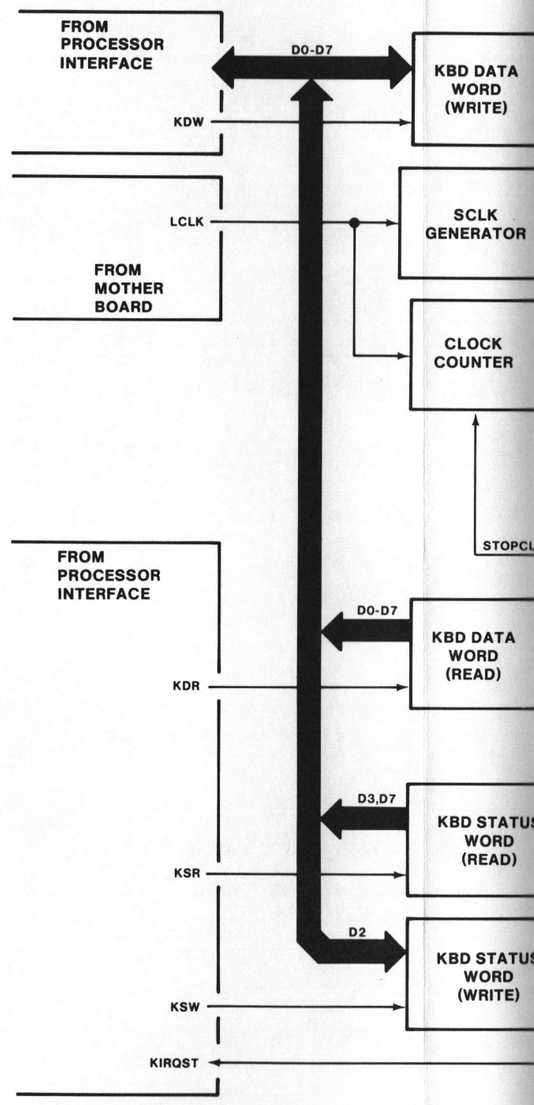
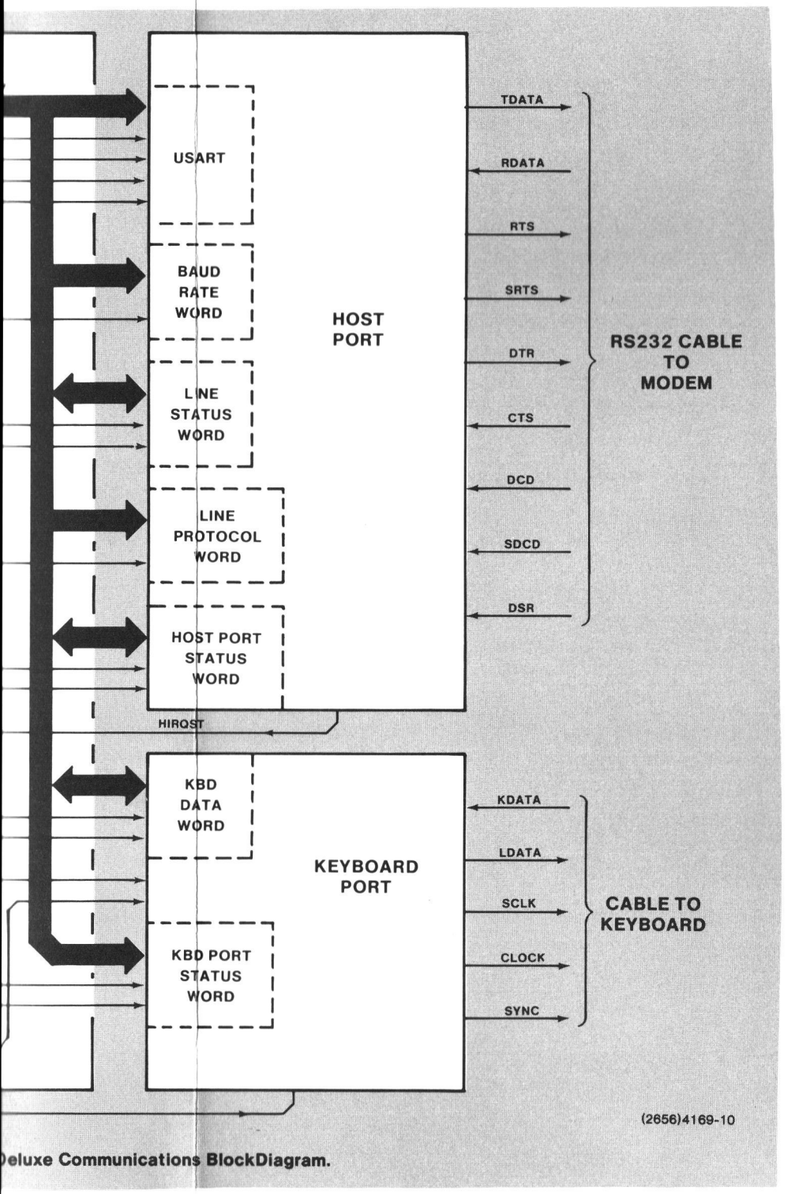


Figure 5-7. Deluxe Communications BlockDiagram.





Deluxe Communications BlockDiagram.

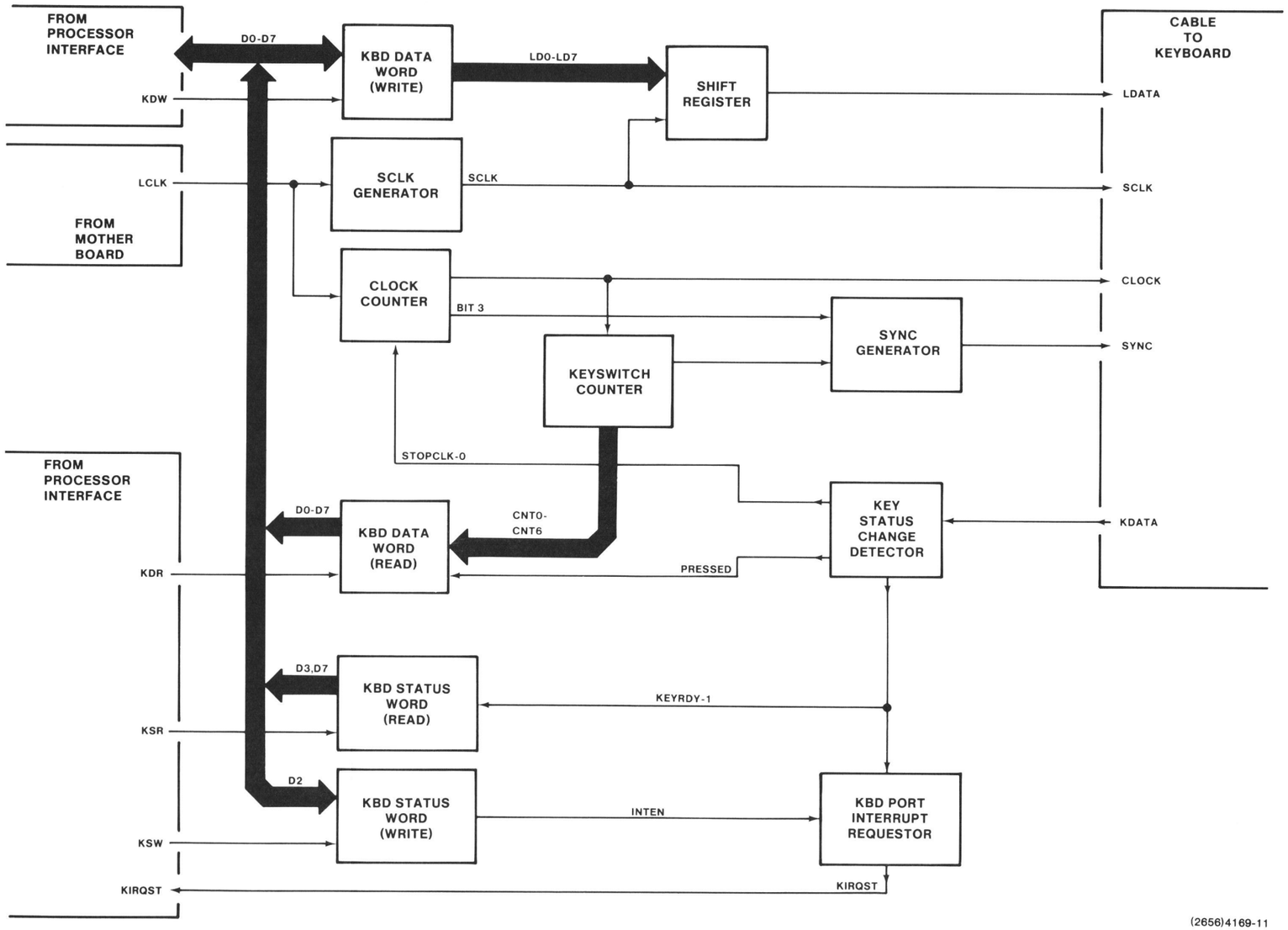


Figure 5-8. Keyboard Port Block Diagram.

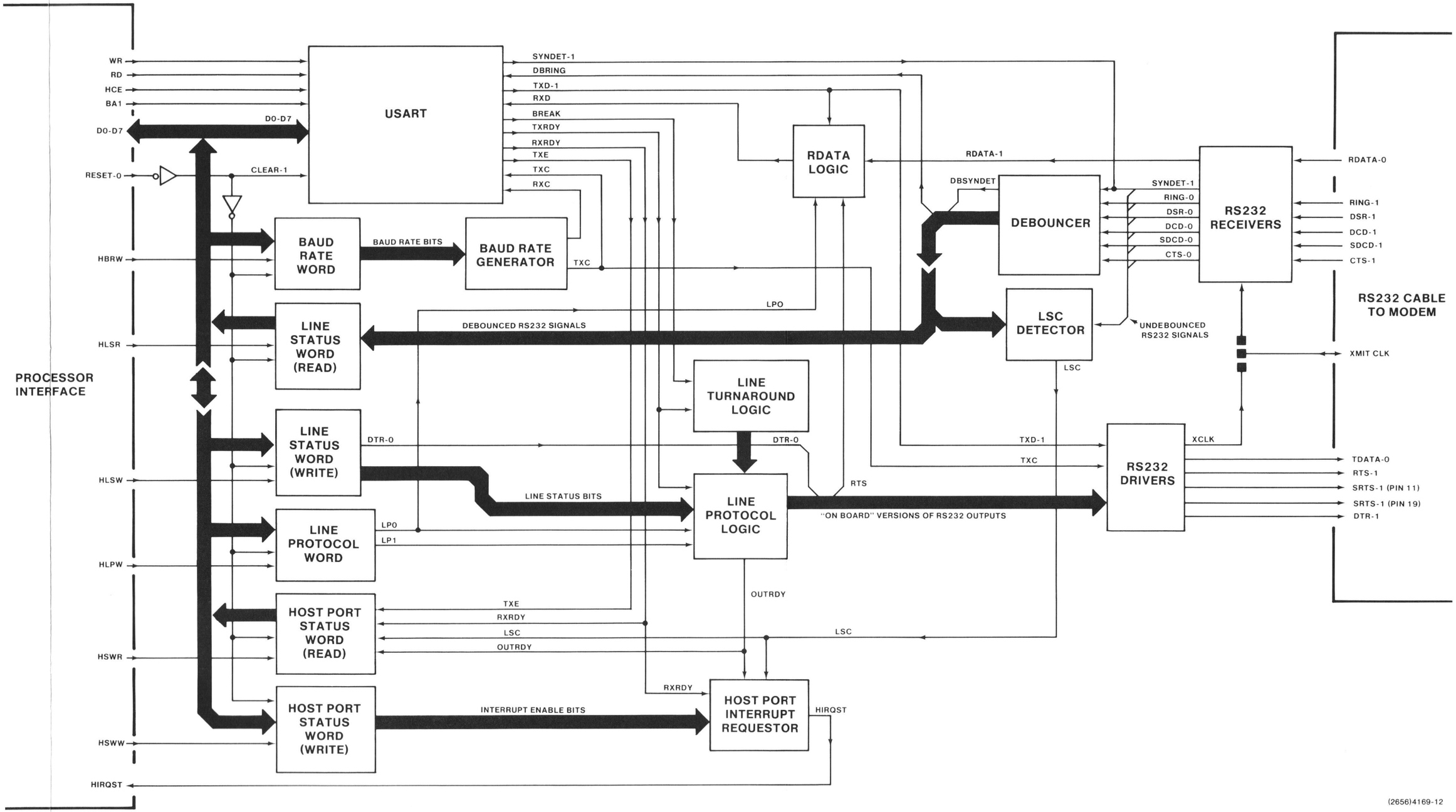


Figure 5-9. Host Port Block Diagram.

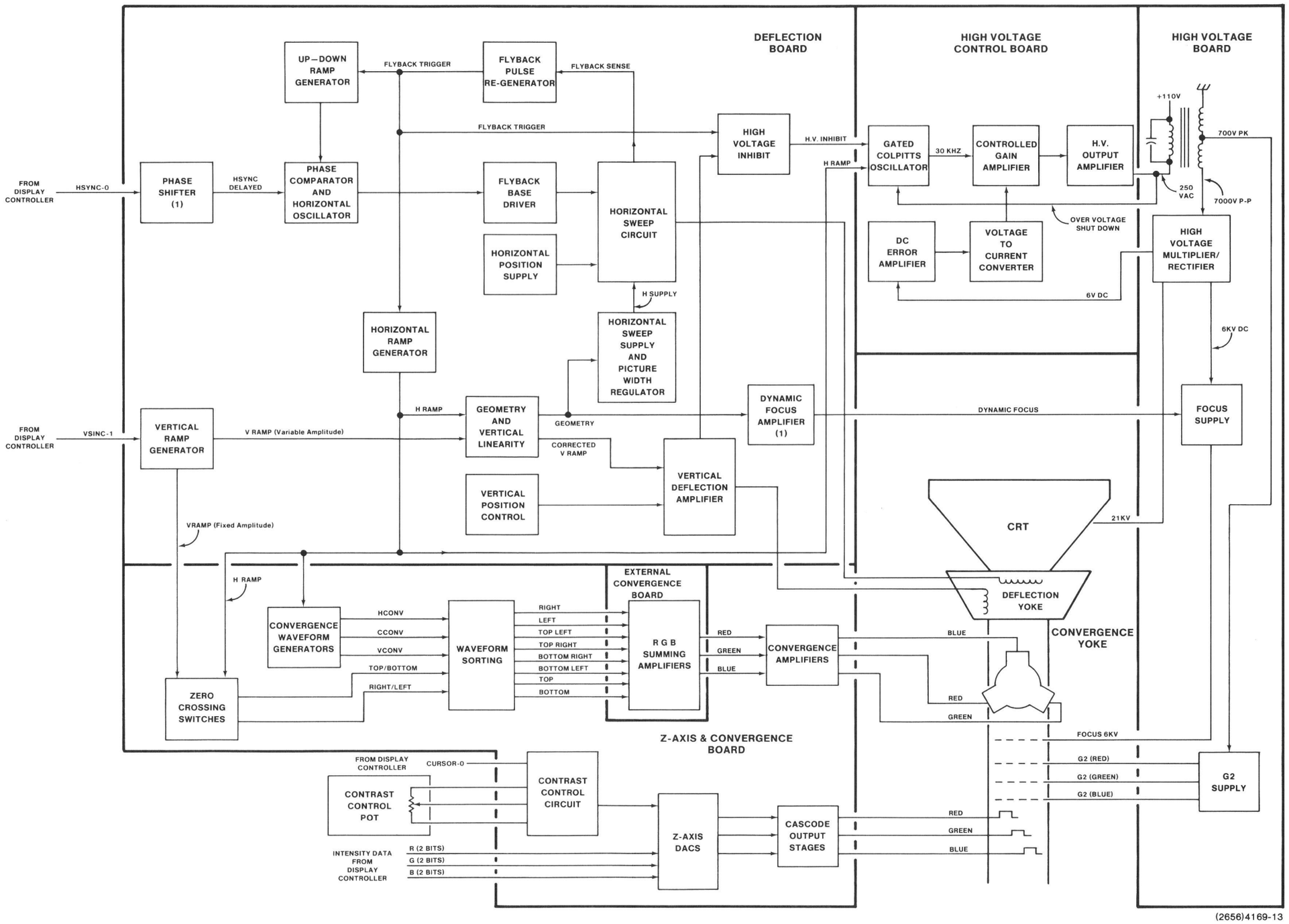


Figure 5-10. Color Video Display Block Diagram.

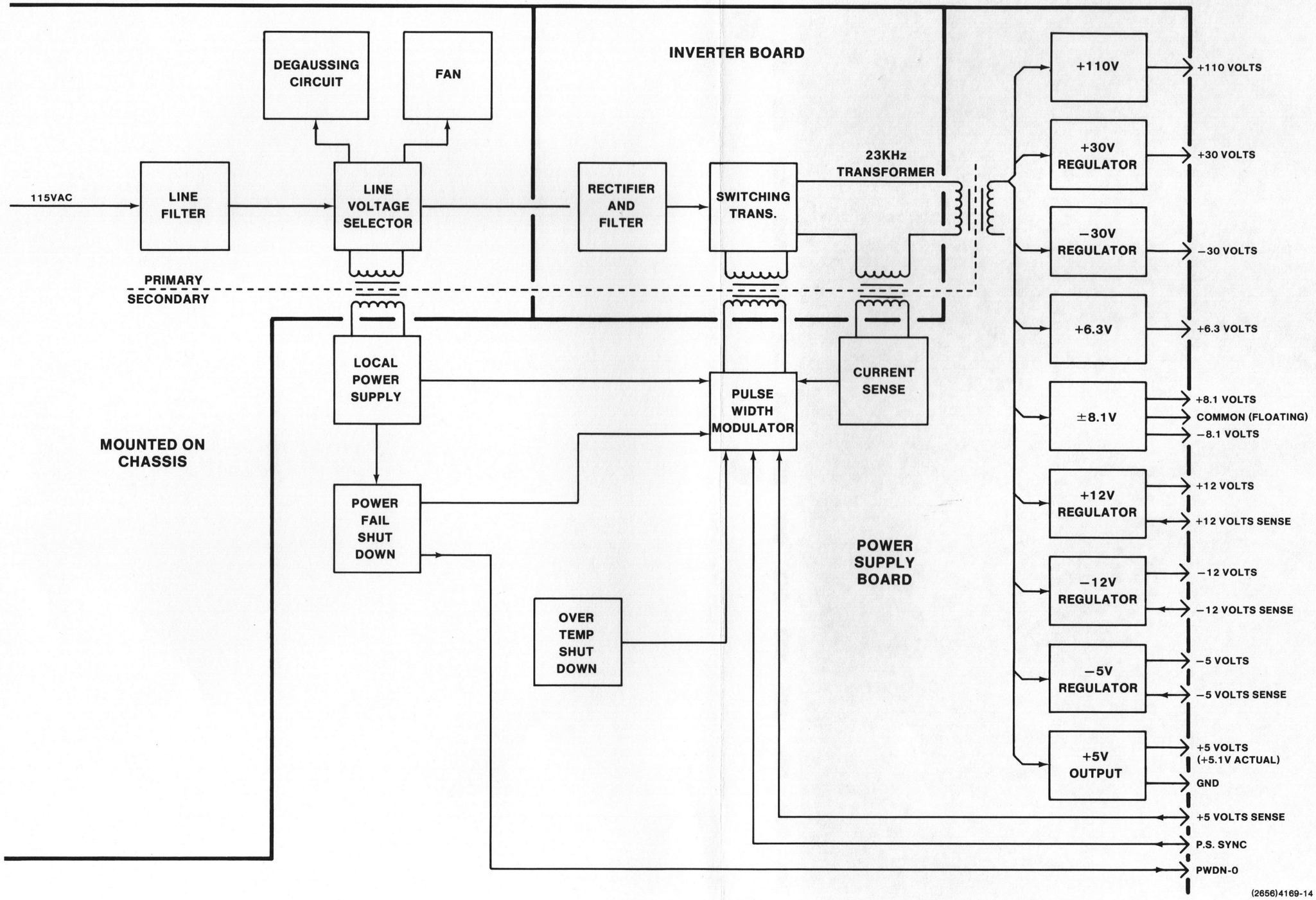


Figure 5-11. Power Supply Block Diagram.

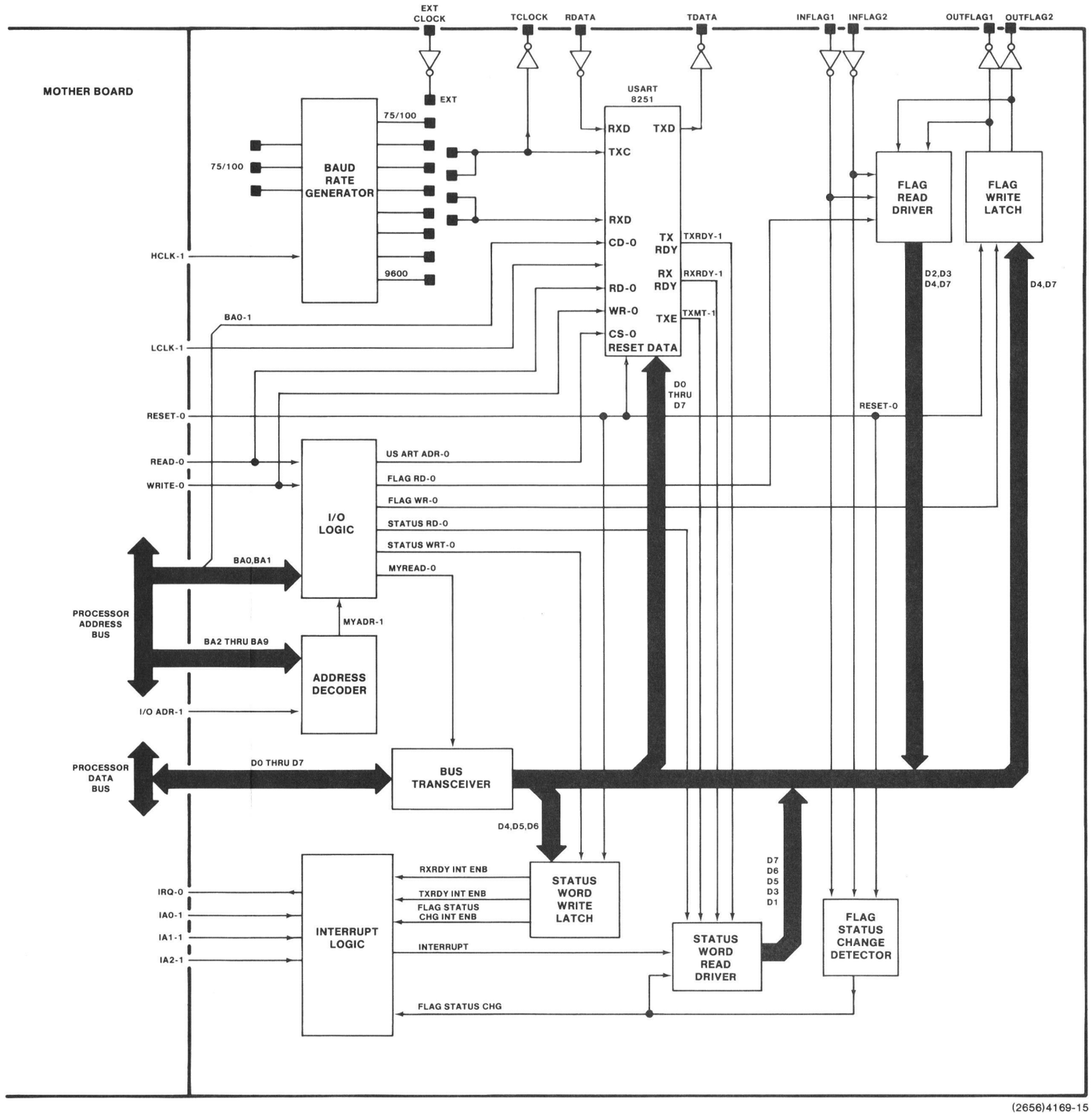
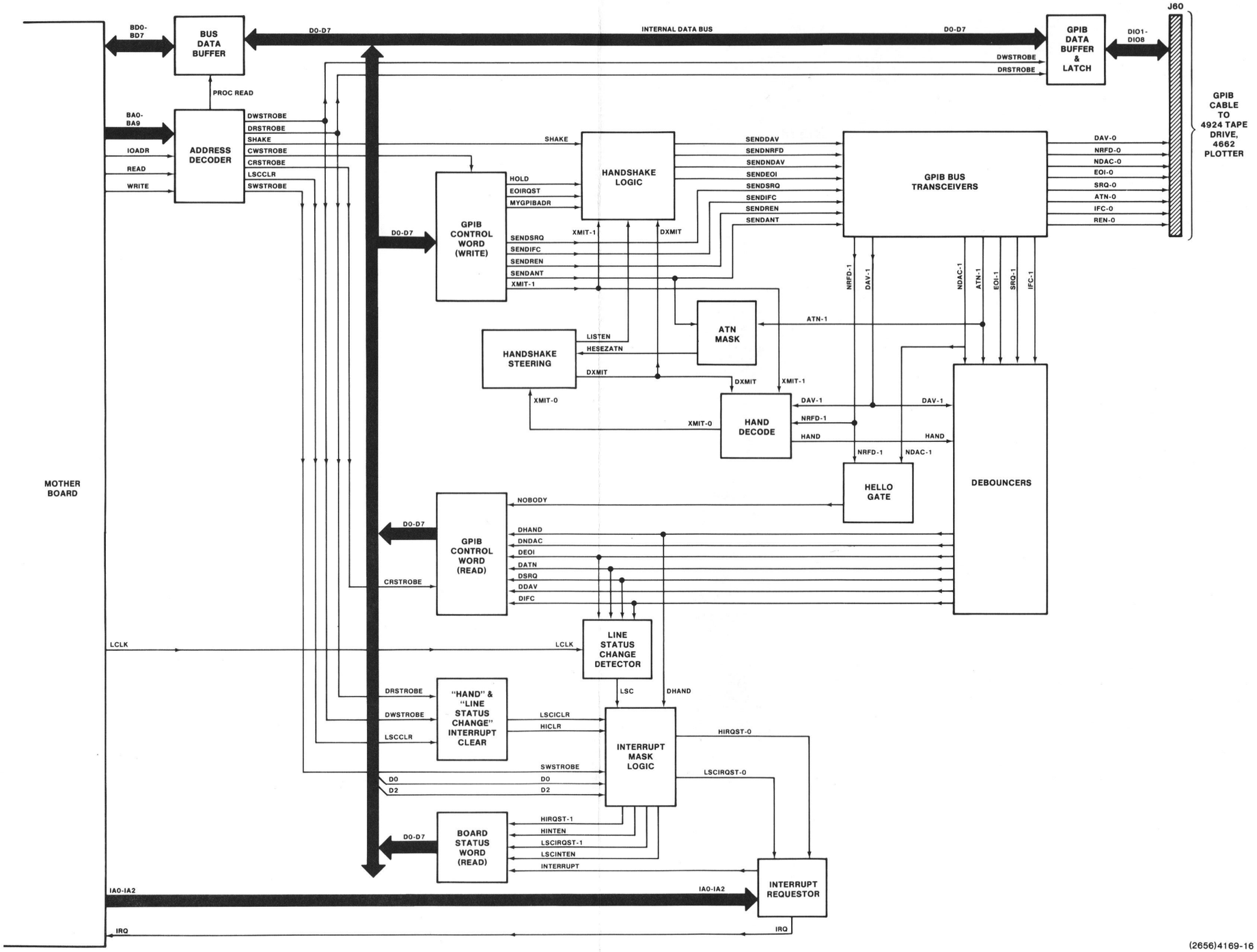


Figure 5-12. RS-232 Peripheral Interface Block Diagram.

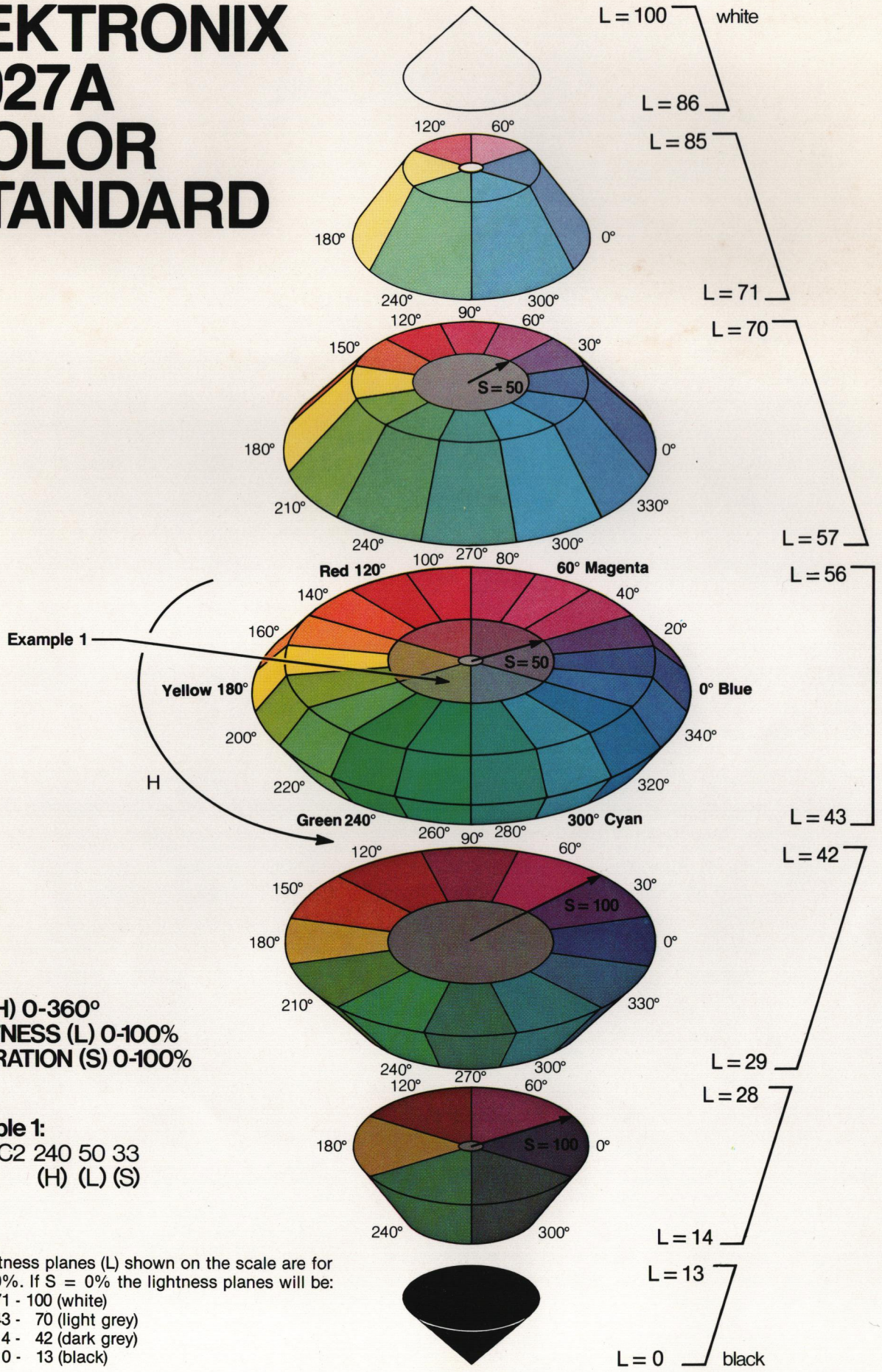


(2656)4169-16

Figure 5-13. GPIB Peripheral Interface Block Diagram.

Appendix A

TEKTRONIX 4027A COLOR STANDARD



HUE (H) 0-360°
LIGHTNESS (L) 0-100%
SATURATION (S) 0-100%

Example 1:
!MAP C2 240 50 33
(H) (L) (S)

***Note:**
The lightness planes (L) shown on the scale are for S = 100%. If S = 0% the lightness planes will be:

L = 71 - 100 (white)
L = 43 - 70 (light grey)
L = 14 - 42 (dark grey)
L = 0 - 13 (black)

Figure A-1. 4027A Color Standard.

TEKTRONIX 4027A COLOR STANDARD

Overview:

The world of color is filled with ambiguous terminology, i.e. intensity, purity, value, etc. Many color users feel that "color theory" is a prerequisite to operating color systems; T.V., Videotaping, Photography, Computer Graphics.

In order to end this confusion, Tektronix has developed a color language and function based on human engineering, rather than machine engineering. Below is a description of this system, which will provide a clear and concise means for understanding how color is defined and how our syntax was derived.

4027A Color Concepts:

Color selection is specified by hue, lightness and saturation which is the HLS method. The definitions are as follows:

Hue: The characteristic associated with a color name such as red, yellow, green, blue, etc. Hue is a gradation of color advanced by degrees, thus represented as an angle from 0 to 360.

Lightness: The characteristic that allows the color to be ranked on a scale from dark to light. Lightness is expressed as a parameter ranging from 0 to 100% with black being 0 (bottom of cone) and white being 100% (top of cone).

Saturation: The characteristic which describes the extent to which a color differs from a gray of the same lightness. Saturation is expressed as percentage, ranging from 0% (maximum white content at that lightness level) to 100% (full saturated).

Geometrically, colors can be described in terms of a double cone (see Figure 1). Variations in lightness are represented along the axis, with white at the apex of the cone and black at the opposite apex. Variations in saturation are represented by radial distances from the lightness axis, in constant lightness planes. Hue is represented as an angular quantity from a known reference point.

The 64 colors available in the 4027A are discrete samples from this continuous color space. They are obtained by intersecting the cone into several planes of constant lightness.

A better understanding of the color standard can be had by looking at a cross section of the double-ended cone (Figure A-2). There are four gray levels along the middle of the cone. At 0% saturation the four levels of gray are black, dark gray, light gray, and white. At any other value of saturation, different hues (color mixtures) are obtained. Hue has no effect at 0% saturation. A maximum of seven different "planes" of color can be obtained at any value of saturation except 0%.

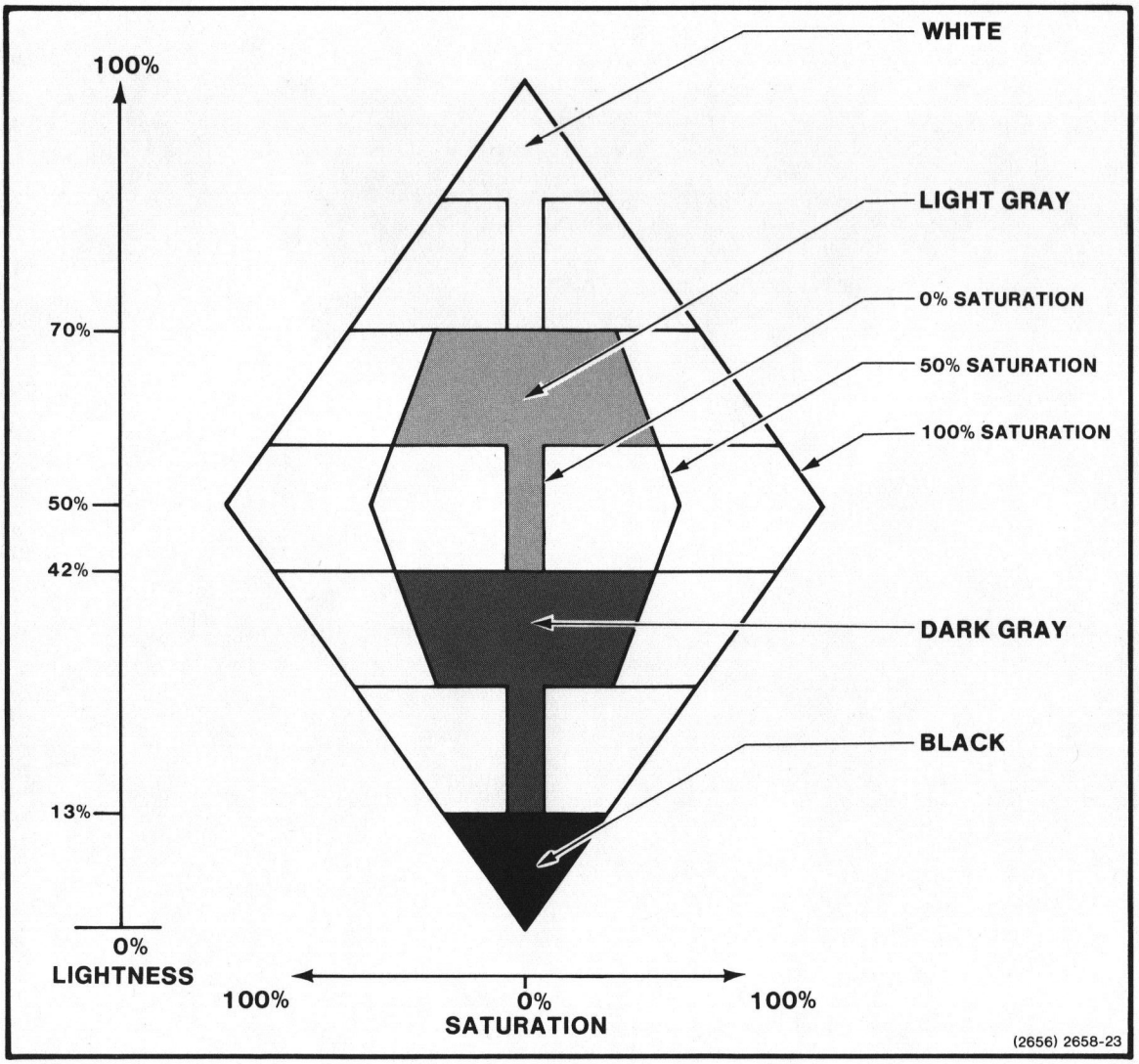


Figure A-2. Cross Section of the 4027A Color Standard.